

# Clusters and features from combinatorial stochastic processes

Tamara Broderick, Michael I. Jordan, Jim Pitman

March 8, 2013

## Abstract

In partitioning—a.k.a. clustering—data, we associate each data point with one and only one of some collection of groups called clusters or partition blocks. Here, we formally develop an analogous problem, called feature allocation, for associating data points with arbitrary non-negative integer numbers of groups, now called features or topics. We review known combinatorial stochastic process representations of clustering and develop analogous representations for the feature allocation case. We illustrate the clustering representations with examples that include the canonical nonparametric Bayesian clustering prior: the Chinese restaurant process or Dirichlet process. We not only illustrate the feature allocation representations with the canonical nonparametric Bayesian feature prior—the Indian buffet process or beta process—but also simultaneously discover new connections between the different representations for the Indian buffet process. We thereby bring the same level of completeness to the treatment of the Indian buffet that has previously been developed for the Chinese restaurant.

## 1 Introduction

Bayesian nonparametrics is the area of Bayesian analysis in which the finite-dimensional prior distributions of classical Bayesian analysis are replaced with stochastic processes. While the rationale for allowing infinite collections of random variables into Bayesian inference is often taken to be that of diminishing the role of prior assumptions, it is also possible to view the move to nonparametrics as supplying the Bayesian paradigm with a richer collection of distributions with which to express prior belief, thus in some sense emphasizing the role of the prior. In practice, however, the field has been dominated by two stochastic processes—the Gaussian process and the Dirichlet process—and thus the flexibility promised by the nonparametric approach has arguably not yet been delivered. In the current paper we aim to provide a broader perspective on the kinds of stochastic processes that can provide a useful toolbox for Bayesian nonparametric analysis. Specifically, we focus on *combinatorial stochastic processes*

as embodying mathematical structure that is useful for both model specification and inference.

The phrase “combinatorial stochastic process” comes from probability theory [Pitman, 2006], where it refers to connections between stochastic processes and the mathematical field of combinatorics. Indeed, the focus in this area of probability theory is on random versions of classical combinatorial objects such as partitions, trees, and graphs—and on the role of combinatorial analysis in establishing properties of these processes. As we wish to argue, this connection is also fruitful in a statistical setting. Roughly speaking, in statistics it is often natural to model observed data as arising from a combination of underlying factors. In the Bayesian setting, such models are often embodied as latent variable models in which the latent variable has a compositional structure. Making explicit use of ideas from combinatorics in latent variable modeling can not only suggest new modeling ideas, but it can also provide essential help with calculations of marginal and conditional probability distributions.

The Dirichlet process already serves as one interesting exhibit of the connections between Bayesian nonparametrics and combinatorial stochastic processes. On the one hand, the Dirichlet process is classically defined in terms of a partition of a probability space [Ferguson, 1973], and there are many well-known connections between the Dirichlet process and urn models [Blackwell and MacQueen, 1973, Hoppe, 1984]. In the current paper, we will review and expand upon some of these connections, beginning our treatment (non-traditionally) with the notion of an *exchangeable partition probability function* (EPPF) and, from there, discussing related urn models, stick-breaking representations, subordinators, and random measures.

On the other hand, the Dirichlet process is limited in terms of the statistical notion of “combination of underlying factors” that we referred to above. Indeed, the Dirichlet process is generally used in a statistical setting to express the idea that each data point is associated with one and only one underlying factor. In contrast to such *clustering models*, we wish to also consider *featural models*, where each data point is associated with a set of underlying features and it is the interaction among these features that gives rise to an observed data point. Focusing on the case in which these features are binary, we develop some of the combinatorial stochastic process machinery needed to specify featural priors. Specifically, we develop a counterpart to the EPPF, which we refer to as the *exchangeable feature probability function* (EFPF), that characterizes the combinatorial structure of certain featural models. We again develop connections between this combinatorial function and suite of related stochastic processes, including urn models, stick-breaking representations, subordinators, and random measures. As we will discuss, a particular underlying random measure in this case is the *beta process*, originally studied by Hjort [1990] as a model of random hazard functions in survival analysis, but adapted by Thibaux and Jordan [2007] for applications in featural modeling.

For statistical applications it is not enough to develop expressive prior specifications, but it is also essential that inferential computations involving the posterior distribution are tractable. One of the reasons for the popularity of the

Dirichlet process is that the associated urn models and stick-breaking representations yield a variety of useful inference algorithms [Neal, 2000]. As we will see, analogous algorithms are available for featural models. Thus, as we discuss each of the various representations associated with both the Dirichlet process and the beta process, we will also (briefly) discuss some of the consequences of each for posterior inference.

The remainder of the paper is organized as follows. We start by reviewing partitions and introducing feature allocations in Section 2 in order to define distributions over these models (Section 3) via the exchangeable partition probability function (EPPF) in the partition case (Section 3.1) and the exchangeable feature probability function (EFPPF) in the feature allocation case (Section 3.2). Illustrating these exchangeable probability functions with examples, we will see that the well-known *Chinese restaurant process* (CRP) [Blackwell and MacQueen, 1973, Aldous, 1985] corresponds to a particular EPPF choice (Example 1) and the *Indian buffet process* (IBP) [Griffiths and Ghahramani, 2006] corresponds to a particular choice of EFPPF (Example 5). From here, we progressively build up richer models by first reviewing stick lengths (Section 4), which we will see represent limiting frequencies of certain clusters or features, and then subordinators (Section 5), which further associate a random label with each cluster or feature. We illustrate these progressive augmentations on the CRP (Examples 1, 6, 10, 18, and 20) and IBP examples (Examples 5, 7, 11, and 15). We augment the model once more to obtain a random measure on a general space of cluster or feature parameters in Section 6; here, among other relations, we find that the CRP example is the marginal of a Dirichlet process (Example 23), and the IBP example is the marginal of a beta process (Example 24). Finally, in Section 7, we mention some of the other combinatorial stochastic processes, beyond the Dirichlet process and the beta process, that have begun to be studied in the Bayesian nonparametrics literature, and we provide suggestions for further developments.

## 2 Partitions and feature allocations

While we have some intuitive ideas about what constitutes a mixture or admixture model, we want to formalize these ideas before proceeding. We begin with the underlying combinatorial structure on the data indices. We think of  $[N] := \{1, \dots, N\}$  as representing the indices of the first  $N$  data points. There are different groupings that we apply in the mixture case (*partitions*) and admixture case (*feature allocations*); we describe these below.

First, we wish to describe the space of *partitions* over the indices  $[N]$ . In particular, a partition  $\pi_N$  of  $[N]$  is defined to be a collection of mutually exclusive, mutually exhaustive, non-empty subsets of  $[N]$  called *blocks*; that is,  $\pi_N = \{A_1, \dots, A_K\}$  for some number of partition blocks  $K$ . An example partition of  $[6]$  is  $\pi_6 = \{\{1, 3, 4\}, \{2\}, \{5, 6\}\}$ . Similarly, a partition of  $\mathbb{N} = \{1, 2, \dots\}$  is a collection of mutually exclusive, mutually exhaustive non-empty subsets of  $\mathbb{N}$ . In this case, the number of blocks may be infinite, and

we have  $\pi_N = \{A_1, A_2, \dots\}$ . An example partition of  $\mathbb{N}$  into two blocks is  $\{\{n : n \text{ is even}\}, \{n : n \text{ is odd}\}\}$ .

We introduce a generalization of a partition called a *feature allocation* that relaxes both the mutually exclusive and mutually exhaustive restrictions. In particular, a feature allocation  $f_N$  of  $[N]$  is defined to be a multiset of non-empty subsets of  $[N]$ , again called *blocks*, such that no index  $n$  belongs to infinitely many blocks. We write  $f_N = \{A_1, \dots, A_K\}$  for some number of feature allocation blocks  $K$ . An example feature allocation of  $[6]$  is  $f_6 = \{\{2, 3\}, \{2, 4, 6\}, \{3\}, \{3\}, \{3\}\}$ . Just as the blocks of a partition are sometimes called *clusters*, so are the blocks of a feature allocation sometimes called *features*. We note that a partition is always a feature allocation, but the converse statement does not hold in general; for instance,  $f_6$  given above is not a partition.

In the remainder of this section, we continue our development in terms of feature allocations since partitions are a special case of the former object. We note that we can extend the idea of random partitions [Aldous, 1985] to consider *random feature allocations*. If  $\mathcal{F}_N$  is the space of all feature allocations of  $[N]$ , then a random feature allocation  $F_N$  of  $[N]$  is a random element of this space.

We next introduce a few useful assumptions on our random feature allocation. Just as exchangeability of observations is often a central assumption in statistical modeling, so will we make use of *exchangeable feature allocations*. To rigorously define such feature allocations, we introduce the following notation. Let  $\sigma : \mathbb{N} \rightarrow \mathbb{N}$  be a finite permutation. That is, for some finite value  $N_\sigma$ , we have  $\sigma(n) = n$  for all  $n > N_\sigma$ . Further, for any block  $A \subset \mathbb{N}$ , denote the permutation applied to the block as follows:  $\sigma(A) := \{\sigma(n) : n \in A\}$ . For any feature allocation  $F_N$ , denote the permutation applied to the feature allocation as follows:  $\sigma(F_N) := \{\sigma(A) : A \in F_N\}$ . Finally, let  $F_N$  be a random feature allocation of  $[N]$ . Then we say that  $F_N$  is exchangeable if  $F_N \stackrel{d}{=} \sigma(F_N)$  for every finite permutation  $\sigma$ .

Our second assumption in what follows will be that we are dealing with a *consistent* feature allocation. We often implicitly imagine the indices arriving one at a time: first 1, then 2, up to  $N$  or beyond. We will find it useful, similarly, in defining random feature allocations to suppose that the randomness at stage  $n$  somehow agrees with the randomness at stage  $n + 1$ . More formally, we say that a feature allocation  $f_M$  of  $[M]$  is a *restriction* of a feature allocation  $f_N$  of  $[N]$  for  $M < N$  if

$$f_M = \{A \cap [M] : A \in f_N\}.$$

Let  $\mathcal{R}_N(f_M)$  be the set of all feature allocations of  $[N]$  whose restriction to  $[M]$  is  $f_M$ . Then we say that the sequence of random feature allocations  $(F_n)$  is *consistent* if (1) there is some function  $q$  such that  $q(f_N)$  is the probability that  $F_N$  takes the value  $f_N$  for each  $N$  and (2) moreover, for all  $M$  and  $N$  such that  $M < N$ , we have

$$q(f_M) = \sum_{f_N \in \mathcal{R}_N(f_M)} q(f_N). \quad (1)$$

With this consistency condition in hand, we can define a random feature allocation  $F_\infty$  of  $\mathbb{N}$ . In particular, such a feature allocation is characterized by the sequence of consistent finite restrictions  $F_N$  to  $[N]$ :  $F_N := \{A \cap [N] : A \in F_\infty\}$ . Then  $F_\infty$  is equivalent to a consistent sequence of finite feature allocations and may be thought of as a random element of the space of such sequences:  $F_\infty = (F_n)_n$ . We let  $\mathcal{F}_\infty$  denote the space of consistent feature allocations, of which each random feature allocation is a random element, and we see that the sigma-algebra associated with this space is generated by the finite-dimensional sigma-algebras of the restricted random feature allocations  $F_n$ .

We say that  $F_\infty$  is exchangeable if  $F_\infty \stackrel{d}{=} \sigma(F_\infty)$  for every finite permutation  $\sigma$ . That is, when the permutation  $\sigma$  changes no indices above  $N$ , we require  $F_N \stackrel{d}{=} \sigma(F_N)$ , where  $F_N$  is the restriction of  $F_\infty$  to  $[N]$ .

In what follows, we consider exchangeable, consistent random partitions and feature allocations.

### 3 Exchangeable probability functions

Once we know that we can construct (exchangeable and consistent) random partitions and feature allocations, it remains to find useful ways of representing the distributions  $q$ , as in Eq. (1), over these objects.

#### 3.1 Exchangeable partition probability function

Consider first an exchangeable, consistent, random partition  $(\Pi_n)$ . From Eq. (1), we have a function  $q$  describing the distribution of the partition. By the exchangeability assumption, this distribution should depend only on the (unordered) sizes of the blocks. Therefore, there is further a function  $p$  that is symmetric in its arguments such that, for any specific partition assignment  $\pi_n = \{A_1, \dots, A_K\}$ , we have

$$q(\Pi_n = \pi_n) = p(|A_1|, \dots, |A_K|).$$

The function  $p$  is called the *exchangeable partition probability function* (EPPF) [Pitman, 1995].

**Example 1** (Chinese restaurant process). The Chinese restaurant process (CRP) [Blackwell and MacQueen, 1973] is an iterative description of a partition via the conditional distribution of increasing partition index labels. The Chinese restaurant metaphor forms an equivalence between customers entering a Chinese restaurant and partition indices; customers who share a table at the restaurant represent indices belonging to the same partition block.

To generate the first index label, the first customer enters the restaurant and sits down at some table, necessarily unoccupied since no one else is in the restaurant. A “dish” is set out at the new table; call the dish “1” since it is the first dish. The customer is assigned the label of the dish at her table:

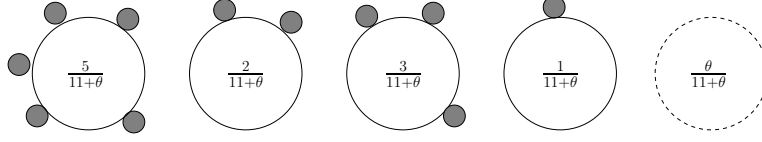


Figure 1: The diagram represents a possible CRP seating arrangement after 11 customers have entered a restaurant with parameter  $\theta$ . Each large white circle is a table, and the smaller gray circles are customers sitting at those tables. If a 12th customer enters, the expressions in the middle of each table give the probability of the new customer sitting there. In particular, the probability of the 12th customer sitting at the first table is  $5/(11 + \theta)$ , and the probability of the 12th customer forming a new table is  $\theta/(11 + \theta)$ .

$Z_1 = 1$ . Recursively, for a restaurant with *concentration parameter*  $\theta$ , the  $n$ th customer sits at an occupied table with probability in proportion to the number of people at the table and at a new table with probability proportional to  $\theta$ . In the former case,  $Z_n$  takes the value of the existing dish at the table, and in the latter case, the next available dish  $k$  (equal to the number of existing tables plus one) appears at the new table, and  $Z_n = k$ . By summing over all possibilities when the  $n$ th customer arrives, one obtains the normalizing constant for the distribution across potential occupied tables:  $(n - 1 + \theta)^{-1}$ . To summarize, if we let  $K_n := \max\{Z_1, \dots, Z_n\}$ , then the distribution of table assignments for the  $n$ th customer is

$$\begin{aligned} \mathbb{P}(Z_n = k | Z_1, \dots, Z_{n-1}) \\ = (n - 1 + \theta)^{-1} \begin{cases} \#\{m : m < n, Z_m = j\} & \text{for } j \leq K_{n-1} \\ \theta & \text{for } k = K_{n-1} + 1 \end{cases} \end{aligned} \quad (2)$$

We note that an equivalent generative description follows a Polya urn style in specifying that each incoming customer sits next to an existing customer with probability proportional to 1 and forms a new table with probability proportional to  $\theta$  [Hoppe, 1984].

Next, we find the probability of the partition induced by considering the collection of indices sitting at each table as a block in the partition. Suppose that the set of cardinalities of non-zero table occupancies is  $\{N_1, \dots, N_K\}$  with  $N := \sum_{k=1}^K N_k$ . That is, we are considering the case when  $N$  customers have entered the restaurant and sat at  $K$  different tables in the specified configuration. We can see from Eq. (2) that when the  $n$ th customer enters ( $n > 1$ ), we obtain a factor of  $n - 1 + \theta$  in the denominator. Using the following notation for the rising and falling factorial

$$x_{M \uparrow a} := \prod_{m=0}^{M-1} (x + ma), \quad x_{M \downarrow a} := \prod_{m=0}^{M-1} (x - ma),$$

we find that the denominator must be  $(\theta + 1)_{N-1 \uparrow 1}$ . Similarly, each time a customer forms a new table except for the first table, we obtain a factor of  $\theta$  in the

numerator. Combining these factors, we find a factor of  $\theta^{K-1}$  in the numerator. Finally, each time a customer sits at an existing table with  $n$  occupants, we obtain a factor of  $n$  in the numerator. Thus, for each table  $k$ , we have a factor of  $(N_k - 1)!$  once all customers have entered the restaurant. Having collected all terms in the process, we see that the probability of the resulting configuration is:

$$q(\Pi_N = \pi_N) = \frac{\theta^{K-1} \prod_{k=1}^K (N_k - 1)!}{(\theta + 1)_{N-1 \uparrow 1}}. \quad (3)$$

We first note that Eq. (3) depends only on the block sizes and not on the order of arrival of the customers or dishes at the tables. We conclude that the partition generated according to the CRP scheme is exchangeable. Moreover, as the partition  $\Pi_M$  is the restriction of  $\Pi_N$  to  $[M]$  for any  $N > M$  by construction, we have that Eq. (3) satisfies the consistency condition of Eq. (1). It follows that Eq. (3) is, in fact, an EPPF. ■

### 3.2 Exchangeable feature probability function

Just as we considered an exchangeable, consistent, random partition above, so we now turn to an exchangeable, consistent, random feature allocation  $(F_n)$  now. Let  $f_N = \{A_1, \dots, A_K\}$  be any particular feature allocation. In calculating  $q(F_N = f_N)$ , we start by demonstrating in the next example that this probability in some sense undercounts features when they contain exactly the same indices: e.g.,  $A_j = A_k$  for some  $j \neq k$ . For instance, consider the following example.

**Example 2** (A two-block, Bernoulli feature allocation). Let  $q_A, q_B \in (0, 1)$  represent the frequencies of features  $A$  and  $B$ . Draw  $Z_{A,n} \stackrel{iid}{\sim} \text{Bern}(q_A)$  and  $Z_{B,n} \stackrel{iid}{\sim} \text{Bern}(q_B)$ , independently. Construct the random feature allocation by collecting those indices with successful draws:

$$F_N := \{\{n : n \leq N, Z_{A,n} = 1\}, \{n : n \leq N, Z_{B,n} = 1\}\}.$$

Then the probability of the feature allocation  $F_5 = f_5 := \{\{2, 3\}, \{2, 3\}\}$  is

$$q_A^2(1 - q_A)^3 q_B^2(1 - q_B)^3,$$

but the probability of the feature allocation  $F_5 = f'_5 := \{\{2, 3\}, \{2, 5\}\}$  is

$$2q_A^2(1 - q_A)^3 q_B^2(1 - q_B)^3.$$

The difference is that in the latter case the features can be distinguished, and so we must account for the two possible pairings of features to frequencies  $\{q_A, q_B\}$ .

Now, instead, let  $\tilde{F}_N$  be  $F_N$  with a uniform random ordering on the features. There is just a single possible ordering of  $f_5$ , so the probability of  $\tilde{F}_N = \tilde{f}_5 := (\{2, 3\}, \{2, 3\})$  is again

$$q_A^2(1 - q_A)^3 q_B^2(1 - q_B)^3.$$

However, there are two orderings of  $f'_5$ , so the probability of  $\tilde{F}_N = \tilde{f}'_5 := (\{2, 5\}, \{2, 3\})$  is

$$q_A^2(1 - q_A)^3 q_B^2(1 - q_B)^3,$$

and the same holds for the other ordering.  $\blacksquare$

For reasons suggested by the previous example, we will find it useful to work with the random feature allocation after uniform random ordering,  $\tilde{F}_N$ . One way to achieve such an ordering and maintain consistency across different  $N$  is to associate some independent, continuous random variable with each feature; e.g. assign a uniform random variable on  $[0, 1]$  to each feature and order the features according to the order of the assigned random variables. When we view feature allocations constructed as marginals of a *subordinator* in Section 5, we will see that this construction is natural.

In general, given a probability of a random feature allocation,  $q(F_N = f_N)$ , we can find the probability of a *random ordered feature allocation*,  $q(\tilde{F}_N = \tilde{f}_N)$  as follows. Let  $H$  be the number of unique elements of  $F_N$ , and let  $(\tilde{K}_1, \dots, \tilde{K}_H)$  be the multiplicities of these unique elements in decreasing size. Then

$$q(\tilde{F}_N = \tilde{f}_N) = \binom{K}{\tilde{K}_1, \dots, \tilde{K}_H}^{-1} q(F_N = f_N), \quad (4)$$

where

$$\binom{K}{\tilde{K}_1, \dots, \tilde{K}_H} := \frac{K!}{\tilde{K}_1! \dots \tilde{K}_H!}.$$

We will see in Section 5 that augmentation of an exchangeable partition with a random ordering is also natural. However, the probability of an ordered random partition is not substantively different from the probability of an unordered version since the factor contributed by ordering a partition is always  $1/K!$ , where  $K$  here is the number of partition blocks.

With this framework in place, we can see that some ordered feature allocations have a probability function  $p$  nearly as in Eq. (6) that is, moreover, symmetric in its block-size arguments. Consider again the previous example.

**Example 3** (A two-block, Bernoulli feature allocation (continued)). Consider any  $F_N$  with block sizes  $N_1$  and  $N_2$  constructed as in Example 2. Then

$$\begin{aligned} q(\tilde{F}_N = \tilde{f}_N) &= \frac{1}{2} q_A^{N_1} (1 - q_A)^{N - N_1} q_B^{N_2} (1 - q_B)^{N - N_2} \\ &\quad + \frac{1}{2} q_A^{N_2} (1 - q_A)^{N - N_2} q_B^{N_1} (1 - q_B)^{N - N_1} \\ &= p(N, N_1, N_2), \end{aligned} \quad (5)$$

where  $p$  is some function of the number of indices  $N$  and the block sizes  $(N_1, N_2)$  that we note is symmetric in all arguments after the first. In particular, we see that the order of  $N_1$  and  $N_2$  was immaterial.  $\blacksquare$



We note that in the partition case,  $\sum_{k=1}^K |A_k| = N$ , so  $N$  is implicitly an argument to the EPPF. In the feature case, this summation condition no longer holds, so we make the argument  $N$  explicit in Eq. (5).

However, it is not necessarily the case that such a function, much less a symmetric one, exists for exchangeable feature models—in contrast to the case of exchangeable partitions and the EPPF.

**Example 4** (A general two-block feature allocation). We here describe an exchangeable, consistent random feature allocation whose (ordered) distribution does not depend only on the number of indices  $N$  and the sizes of the blocks of the feature allocation.

Let  $p_1, p_2, p_3, p_4$  be fixed frequencies that sum to one. Let  $Y_n$  represent the collection of features to which index  $n$  belongs. For  $n \in \{1, 2\}$ , choose  $Y_n$  independently and identically according to:

$$Y_n = \begin{cases} \{1\} & \text{with probability } p_1 \\ \{2\} & \text{with probability } p_2 \\ \{1, 2\} & \text{with probability } p_3 \\ \emptyset & \text{with probability } p_4 \end{cases}.$$

We form a feature allocation from these labels as follows. For each label (1 or 2), collect those indices  $n$  with the given label appearing in  $Y_n$  to form a feature.

Now consider two possible outcome feature allocations:  $f_2 = \{\{2\}, \{2\}\}$ , and  $f'_2 = \{\{1\}, \{2\}\}$ . The likelihood of any ordering  $\tilde{f}_2$  of  $f_2$  under this model is

$$\mathbb{P}(\tilde{F}_2 = \tilde{f}_2) = p_1^0 p_2^0 p_3^1 p_4^1.$$

The likelihood of any ordering  $\tilde{f}'_2$  of  $f'_2$  is

$$\mathbb{P}(\tilde{F}_2 = \tilde{f}'_2) = p_1^1 p_2^1 p_3^0 p_4^0.$$

It follows from these two likelihoods that we can choose values of  $p_1, p_2, p_3, p_4$  such that  $\mathbb{P}(\tilde{F}_2 = \tilde{f}_2) \neq \mathbb{P}(\tilde{F}_2 = \tilde{f}'_2)$ . But  $\tilde{f}_2$  and  $\tilde{f}'_2$  have the same block counts and  $N$  value ( $N = 2$ ). So there can be no such symmetric function  $p$ , as in Eq. (5), for this model. ■

When a function  $p$  exists in the form

$$q(\tilde{F}_N = \tilde{f}_N) = p(N, |A_1|, \dots, |A_K|) \quad (6)$$

for some ordered feature allocation  $\tilde{f}_N = (A_1, \dots, A_K)$  such that  $p$  is symmetric in all arguments after the first, we call it the *exchangeable feature probability function* (EFPF). We take care to note that the EPPF is not a special case of the EFPF. Indeed, the EPPF assigns zero probability to any multiset in which an index occurs in more than one element of the multiset whereas the indices in the multiset provide no information to the EFPF; only the sizes of the multiset blocks are relevant in the EFPF case.

We next consider a more complex example of an EFPF.

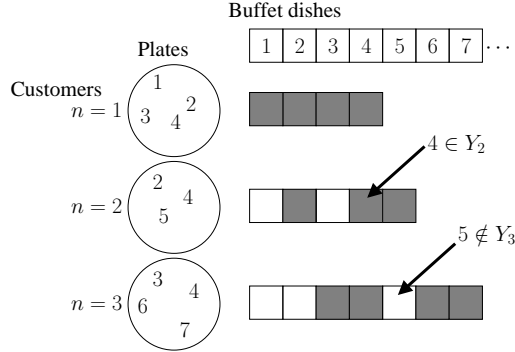


Figure 2: Illustration of an Indian buffet process. The buffet (*top*) consists of a vector of dishes, corresponding to features. Each customer—who enters first decides whether or not to eat dishes that the other customers have already sampled and then tries a random number of new dishes, not previously sampled by any customer. A gray box in position  $(n, k)$  indicates customer  $n$  has sampled dish  $k$ , and a white box indicates the customer has not sampled the dish. In the example, the second customer has sampled exactly those dishes indexed by 2, 4, and 5:  $Y_2 = \{2, 4, 5\}$ .

**Example 5** (Indian buffet process). The Indian buffet process (IBP) [Griffiths and Ghahramani, 2006] is a generative model for a random feature allocation that is specified recursively like the Chinese restaurant process. Also like the CRP, this culinary metaphor forms an equivalence between customers and the indices  $n$  that will be partitioned:  $n \in \mathbb{N}$ . Here, “dishes” again correspond to feature labels just as they corresponded to partition labels for the CRP. But in the IBP case, a customer can sample multiple dishes.

In particular, we start with a single customer, who enters the buffet and chooses  $K_1^+ \sim \text{Pois}(\gamma)$  dishes. Here,  $\gamma > 0$  is called the *mass parameter*, and we will also see the *concentration parameter*  $\theta > 0$  below. None of the dishes have been sampled by any other customers since no other customers have yet entered the restaurant. We label the dishes  $1, \dots, K_1^+$  if  $K_1^+ > 0$ . Recursively, the  $n$ th customer chooses which dishes to sample in two parts. First, for each dish  $k$  that has previously been sampled by any customer in  $1, \dots, n-1$ , customer  $n$  samples dish  $k$  with probability  $N_{n-1,k}/(\theta + n - 1)$  for  $N_{n,k}$  equal to the number of customers indexed  $1, \dots, n$  who have tried dish  $k$ . As each dish represents a feature, and sampling a dish represents that the customer index  $n$  belongs to that feature,  $N_{n,k}$  is the size of the block of the feature labeled  $k$  in the feature allocation of  $[n]$ . Next, customer  $n$  chooses  $K_n^+ \sim \text{Pois}(\theta\gamma/(\theta + n - 1))$  new dishes to try. If  $K_n^+ > 0$ , then the dishes receive unique labels  $K_{n-1} + 1, \dots, K_n$ . Here,  $K_n$  represents the number of sampled dishes after  $n$  customers:  $K_n = K_{n-1} + K_n^+$ .

With this generative model in hand, we can find the probability of a particular feature allocation. We discover its form by enumeration as for the CRP

EPPF in Example 1. At each round  $n$ , we have a Poisson number of new features,  $K_n^+$ , represented. The probability factor associated with these choices is a product of Poisson densities.

$$\prod_{n=1}^N \frac{1}{K_n^+!} \left( \frac{\theta\gamma}{\theta+n-1} \right)^{K_n^+} \exp \left( \frac{\theta\gamma}{\theta+n-1} \right)$$

Let  $M_k$  be the round on which the  $k$ th dish, in order of appearance, is first chosen. Then the denominators for future dish choice probabilities are the factors in the product  $(\theta + M_k) \cdot (\theta + M_k + 1) \cdots (\theta + N - 1)$ . The numerators for the times when the dish is chosen are the factors in the product  $1 \cdot 2 \cdots (N_{N,k} - 1)$ . The numerators for the times when the dish is not chosen yield  $(\theta + M_k - 1) \cdots (\theta + N - 1 - N_{N,k})$ . Let  $A_{n,k}$  represent the collection of indices in the feature with label  $k$  after  $n$  customers have entered the restaurant. Then  $N_{n,k} = |A_{n,k}|$ . Finally, let  $\tilde{K}_1, \dots, \tilde{K}_H$  be the multiplicities of unique feature blocks formed by this model. We note that there are

$$\left[ \prod_{n=1}^N K_n^+! \right] / \left[ \prod_{h=1}^H \tilde{K}_h! \right]$$

rearrangements of the features generated by this process that all yield the same feature allocation. Since they all have the same generating probability, we simply multiply by this factor to find the feature allocation probability. Multiplying all factors together and taking  $f_n = \{A_{N,1}, \dots, A_{N,K_N}\}$  yields

$$\begin{aligned} q(F_N = f_N) &= \frac{\prod_{n=1}^N K_n^+!}{\prod_{h=1}^H \tilde{K}_h!} \cdot \left[ \prod_{n=1}^N \frac{1}{K_n^+!} \left( \frac{\theta\gamma}{\theta+n-1} \right)^{K_n^+} \exp \left( \frac{\theta\gamma}{\theta+n-1} \right) \right] \\ &\cdot \left[ \prod_{k=1}^{K_N} \frac{\Gamma(\theta + M_k)}{\Gamma(\theta + N)} \Gamma(N_{N,k}) \frac{\Gamma(\theta + N - N_{N,k})}{\Gamma(\theta + M_k - 1)} \right] \\ &= \left( \prod_{h=1}^H \tilde{K}_h! \right)^{-1} \left[ (\theta\gamma)^{K_N^+} \exp \left( \frac{\theta\gamma}{\theta+n-1} \right) \right] \cdot \left[ \frac{\prod_{k=1}^{K_N} (\theta + M_k - 1)}{\prod_{n=1}^N (\theta + n - 1)^{K_n^+}} \right] \\ &\cdot \left[ \prod_{k=1}^{K_N} \frac{\Gamma(N_{N,k}) \Gamma(\theta + N - N_{N,k})}{\Gamma(\theta + N)} \right] \\ &= \left( \prod_{h=1}^H \tilde{K}_h! \right)^{-1} (\theta\gamma)^{K_N} \exp \left( -\theta\gamma \sum_{n=1}^N (\theta + n - 1)^{-1} \right) \prod_{k=1}^{K_N} \frac{\Gamma(N_{N,k}) \Gamma(N - N_{N,k} + \theta)}{\Gamma(N + \theta)}. \end{aligned}$$

It follows from Eq. (4) that the probability of a uniform random ordering of the feature allocation is

$$q(\tilde{F}_N = \tilde{f}_N) \tag{7}$$

$$= \frac{1}{K_N!} (\theta\gamma)^{K_N} \exp\left(-\theta\gamma \sum_{n=1}^N (\theta + n - 1)^{-1}\right) \prod_{k=1}^{K_N} \frac{\Gamma(N_{N,k})\Gamma(N - N_{N,k} + \theta)}{\Gamma(N + \theta)}.$$

The distribution of  $\tilde{F}_N$  has no dependence on the ordering of the indices in  $[N]$ . Hence, the distribution of  $F_N$  depends only on the same quantities—the number of indices and the feature block sizes—and the feature multiplicities. So we see that the IBP construction yields an exchangeable random feature allocation. Consistency follows from the recursive construction and exchangeability. Therefore, Eq. (7) is seen to be in EPPF form (cf. Eq. (6)). ■

Above, we have seen two examples of how specifying a conditional distribution for the block membership of index  $n$  given the block membership of indices in  $[n - 1]$  (also called a *prediction rule* [Hansen and Pitman, 1998]) yields an exchangeable probability function: e.g. the EPPF in the CRP case (Example 1) and the EPPF in the IBP case (Example 5). We will see next that the prediction rule can conversely be recovered from the exchangeable probability function specification and therefore the two are equivalent.

### 3.3 Induced allocations and block labeling

In Examples 1 and 5 above, we formed partitions and feature allocations in the following way. For partitions, we assigned labels  $Z_n$  to each index  $n$ . Then we generated a partition of  $[N]$  from the sequence  $(Z_n)_{n=1}^N$  by saying that indices  $m$  and  $n$  are in the same partition block ( $m \sim n$ ) if and only if  $Z_n = Z_m$ . The resulting partition is called the *induced partition* given the labels  $(Z_n)$ . Similarly, given labels  $(Z_n)_{n=1}^\infty$ , we can form an induced partition of  $\mathbb{N}$ . It is easy to check that, given a sequence  $(Z_n)_{n=1}^\infty$ , the induced partitions of the subsequences  $(Z_n)_{n=1}^N$ , will be consistent.

In the feature case, we first assigned label collections  $Y_n$  to each index  $n$ ;  $Y_n$  is interpreted as a set containing the labels of the features to which  $n$  belongs, and we assume it has finite cardinality. In this case, we generate a feature allocation on  $[N]$  from the sequence  $(Y_n)_{n=1}^N$  by first letting  $\{\phi_k\}_{k=1}^K$  be the set of unique values in  $\bigcup_{n=1}^N Y_n$ . Then the features are the collections of indices with shared labels:  $f_N = \{\{n : \phi_k \in Y_n\} : k = 1, \dots, K\}$ . The resulting feature allocation  $f_N$  is called the *induced feature allocation* given the labels  $(Y_n)$ . Similarly, given label collections  $(Y_n)_{n=1}^\infty$ , where each  $Y_n$  has finite cardinality, we can form an induced feature allocation of  $\mathbb{N}$ . As in the partition case, given a sequence  $(Y_n)_{n=1}^\infty$ , we can see that the induced feature allocations of the subsequences  $(Y_n)_{n=1}^N$ , will be consistent.

In reducing to a partition or feature allocation from a set of labels, we shed the information concerning the labels for each partition block or feature. Conversely, we introduce *order of appearance* labeling schemes to give partition blocks or features labels given a partition or feature allocation.

In the partition case, the order of appearance labeling scheme assigns the label 1 to the partition block containing index 1. Recursively, suppose we have seen  $n$  indices in  $k$  different blocks with labels  $\{1, \dots, k\}$ . And suppose the

$n + 1$ st index does not belong to an existing block. Then we assign its block the label  $k + 1$ .

In the feature allocation case, we note that index 1 belongs to  $K_1^+$  features. If  $K_1^+ = 0$ , there are no features to label yet. If  $K_1^+ > 0$ , we assign these  $K_1^+$  features labels in  $\{1, \dots, K_1^+\}$  and note that there are  $K_1^+$  ways of doing so. Unless otherwise specified, we suppose that the labels are chosen uniformly at random. Let  $K_1 = K_1^+$ . Recursively, suppose we have seen  $n$  indices and  $K_n$  different features with labels  $\{1, \dots, K_n\}$ . Suppose the  $n + 1$ st index belongs to  $K_{n+1}^+$  features that have not yet been labeled. If  $K_{n+1}^+ = 0$ , there are no new features to label. If  $K_{n+1}^+ > 0$ , we let  $K_{n+1} = K_n + K_{n+1}^+$  and assign these  $K_{n+1}^+$  features labels in  $\{K_n + 1, \dots, K_{n+1}\}$ , e.g. uniformly at random.

We can use these labeling schemes to find the prediction rule, which makes use of partition block and feature labels, from the EPPF or EFPF. First, consider a partition with EPPF  $p$ . Then, given labels  $(Z_n)_{n=1}^N$  with  $K_N = \max\{Z_1, \dots, Z_N\}$ , we wish to find the distribution of the label  $Z_{N+1}$ . Using an order of appearance labeling, we know that either  $Z_{N+1} \in \{Z_1, \dots, Z_N\}$  or  $Z_{N+1} = K_N + 1$ . Let  $\pi_N = \{A_{N,1}, \dots, A_{N,K_N}\}$  be the partition induced by  $(Z_n)_{n=1}^N$ . Let  $N_{N,k} = |A_{N,k}|$ . So  $N_{N+1,k} = N_k + \mathbb{1}(Z_{N+1} = k)$  for  $k = 1, \dots, K_{N+1}$ , and  $K_{N+1} = K_N + \mathbb{1}\{Z_{N+1} > K_N\}$  is the number of partition blocks in the partition of  $[N + 1]$ ; here, we let  $N_{N,K_{N+1}} = 0$ . Then the conditional distribution satisfies

$$\mathbb{P}(Z_{N+1} = z | Z_1, \dots, Z_N) = \frac{\mathbb{P}(Z_1, \dots, Z_N, Z_{N+1} = z)}{\mathbb{P}(Z_1, \dots, Z_N)}.$$

But the probability of a certain labeling is just the probability of the underlying partition in this construction, so

$$\mathbb{P}(Z_{N+1} = z | Z_1, \dots, Z_N) = \frac{p(N_{N+1,1}, \dots, N_{N+1,K_{N+1}})}{p(N_{N,1}, \dots, N_{N,K_N})}.$$

**Example 6** (Chinese restaurant process). We continue our Chinese restaurant process example by deriving the Chinese restaurant table assignment scheme from the EPPF in Eq. (3). Substituting in the EPPF for the CRP, we find

$$\begin{aligned} & \mathbb{P}(Z_{N+1} = z | Z_1, \dots, Z_N) \\ &= \frac{p(N_{N,1}, \dots, N_{N,K_{N+1}})}{p(N_{N,1}, \dots, N_{N,K_N})} \\ &= \frac{\left(\theta^{K_{N+1}-1} \prod_{k=1}^{K_{N+1}} (N_{N+1,k} - 1)!\right) ((\theta + 1)_{(N+1)-1 \uparrow 1})^{-1}}{\left(\theta^{K_N-1} \prod_{k=1}^{K_N} (N_{N,k} - 1)!\right) ((\theta + 1)_{N-1 \uparrow 1})^{-1}} \\ &= (N + \theta)^{-1} \begin{cases} N_{N,k} & \text{for } z = k \leq K_N \\ \theta & \text{for } z = K_N + 1 \end{cases}, \end{aligned} \tag{8}$$

just as in Eq. (2). ■

To find the feature allocation prediction rule, we now imagine a feature allocation with EFPF  $p$ . Here we must be slightly more careful about counting due to feature multiplicities. Suppose that after  $N$  indices have been seen, we have label collections  $(Y_n)_{n=1}^N$ , containing a total of  $K_N$  features, labeled  $\{1, \dots, K_N\}$ . We wish to find the distribution of  $Y_{N+1}$ . Suppose  $N+1$  belongs to  $K_{N+1}^+$  features that do not contain any index in  $[N]$ . Using an order of appearance labeling, we know that, if  $K_{N+1}^+ > 0$ , the  $K_{N+1}^+$  new features have labels  $K_N + 1, \dots, K_N + K_{N+1}^+$ . Let  $f_N = \{A_1, \dots, A_{K_N}\}$  be the feature allocation induced by  $(Y_n)_{n=1}^N$ . Let  $N_{N,k} = |A_{N,k}|$  be the size of the  $k$ th feature. So  $N_{N+1,k} = N_{N,k} + \mathbb{1}\{k \in Y_{N+1}\}$ , where we let  $N_{K_N+j} = 0$  for all of the features that are first exhibited by index  $N+1$ :  $j \in \{1, \dots, K_{N+1}^+\}$ . Further, let the number of features, including new ones, be written  $K_{N+1} = K_N + K_{N+1}^+$ . Then the conditional distribution satisfies

$$\mathbb{P}(Y_{N+1} = z | Y_1, \dots, Y_N) = \frac{\mathbb{P}(Y_1, \dots, Y_N, Y_{N+1} = z)}{\mathbb{P}(Y_1, \dots, Y_N)}.$$

As we assume that the labels  $Y$  are consistent across  $N$ , the probability of a certain labeling is just the probability of the underlying feature allocation and a combinatorial term accounting for the possible orderings of the new features:

$$\mathbb{P}(Y_{N+1} = z | Y_1, \dots, Y_N) = \frac{1}{K_{N+1}^+!} \frac{p(N_{N+1,1}, \dots, N_{N+1,K_{N+1}})}{p(N_{N,1}, \dots, N_{N,K_N})}. \quad (9)$$

**Example 7** (Indian buffet process). Just as we derived the Chinese restaurant process prediction rule (Eq. (8)) from its EPPF (Eq. (3)) in Example 6, so can we derive the Indian buffet process prediction rule from its EFPF (Eq. (7)) by using Eq. (9). Substituting the IBP EFPF into Eq. (9), we find

$$\begin{aligned} & \mathbb{P}(Z_{N+1} = z | Z_1, \dots, Z_N) \\ &= \frac{1}{K_{N+1}^+!} \frac{(\theta\gamma)^{K_{N+1}} \exp\left(-\theta\gamma \sum_{n=1}^{N+1} (\theta + n - 1)^{-1}\right) \prod_{k=1}^{K_{N+1}} \frac{\Gamma(N_{N+1,k})\Gamma((N+1)-N_{N+1,k}+\theta)}{\Gamma((N+1)+\theta)}}{(\theta\gamma)^{K_N} \exp\left(-\theta\gamma \sum_{n=1}^N (\theta + n - 1)^{-1}\right) \prod_{k=1}^{K_N} \frac{\Gamma(N_{N,k})\Gamma(N-N_{N,k}+\theta)}{\Gamma(N+\theta)}} \\ &= \left[ \frac{1}{K_{N+1}^+!} \exp\left(-\frac{\theta\gamma}{\theta + (N+1) - 1}\right) \cdot \left(\frac{\theta\gamma}{\theta + (N+1) - 1}\right)^{K_{N+1}^+} \right] \\ & \quad \cdot (\theta + (N+1) - 1)^{K_{N+1}^+} \cdot \left[ \prod_{k=K_N+1}^{K_{N+1}} (\theta + (N+1) - 1)^{-1} \right] \\ & \quad \cdot \prod_{k=1}^{K_N} \frac{N_k^{\mathbb{1}\{k \in z\}} (N - N_{N,k} + \theta)^{\mathbb{1}\{k \notin z\}}}{N + \theta} \\ &= \text{Pois}\left(K_{N+1}^+ \mid \frac{\theta\gamma}{\theta + (N+1) - 1}\right) \cdot \prod_{k=1}^{K_N} \text{Bern}\left(\mathbb{1}\{k \in z\} \mid \frac{N_{N,k}}{N + \theta}\right). \end{aligned}$$

The final line is exactly the Poisson likelihood for the number of new features times the Bernoulli likelihoods for the draws of existing features, as described in Example 5. ■

### 3.4 Inference

The prediction rule formulation of the EPPF or EFPPF is particularly useful in providing a means of inferring partitions and feature allocations from a data set. In particular, we assume that we have data points  $X_1, \dots, X_N$  generated in the following manner. In the partition case, we generate an exchangeable, consistent, random partition  $\Pi_N$  according to the distribution specified by some EPPF  $p$ . Next, we assign each partition block a random parameter that characterizes that block. To be precise, for the  $k$ th partition block to appear according to an order of appearance labeling scheme, give this block a new *random* label  $\phi_k \sim H$ , for some continuous distribution  $H$ . For each  $n$ , let  $Z_n = \phi_k$  where  $k$  is the order of appearance label of index  $n$ . Finally, let

$$X_n \stackrel{\text{indep}}{\sim} \hat{F}(Z_n), \quad (10)$$

for some likelihood  $\hat{F}$ . The choices of both  $H$  and  $\hat{F}$  are specific to the problem domain.

Note that the sequence  $(Z_n)_{n=1}^N$  is sufficient to describe the partition  $\Pi_N$  since  $\Pi_N$  is the collection of blocks of  $[N]$  with the same label values  $Z_n$ . The continuity of  $H$  is necessary to guarantee the a.s. uniqueness of the block values. So, if we can describe the posterior distribution of  $(Z_n)_{n=1}^N$ , we can in principle describe the posterior distribution of  $\Pi_N$ .

The posterior distribution of  $(Z_n)_{n=1}^N$  conditional on  $(X_n)_{n=1}^N$  cannot typically be solved for in closed form, so we turn to a method that approximates this posterior. We will see that prediction rules facilitate the design of a Markov Chain Monte Carlo (MCMC) sampler, in which we approximate the desired posterior distribution by a Markov chain of random samples proven to have the true posterior as its equilibrium distribution.

In the Gibbs sampler formulation of MCMC [Geman and Geman, 1984], we sample each parameter in turn and conditional on all other parameters in the model. In our case, we will sequentially sample each element of  $(Z_n)_{n=1}^N$ . The key observation here is that  $(Z_n)_{n=1}^N$  is an exchangeable sequence. This observation follows by noting that the partition is exchangeable by assumption, and the sequence  $(\phi_k)$  is exchangeable since it is iid;  $(Z_n)$  is an exchangeable sequence since it is a function of  $(\Pi_n)$  and  $(\phi_k)$ . Therefore, the distribution of  $Z_n$  given the remaining elements  $\mathbf{Z}_{-n} := (Z_1, \dots, Z_{n-1}, Z_{n+1}, \dots, Z_N)$  is the same as if we thought of  $Z_n$  as the final,  $N$ th element in a sequence with  $N - 1$  preceding values given by  $\mathbf{Z}_{-n}$ . And the distribution of  $Z_N$  given  $\mathbf{Z}_{-N}$  is provided by the prediction rule. The full details of the Gibbs sampler for the CRP in Examples 1 and 6 were introduced by Escobar [1994], MacEachern [1994], Escobar and West [1995] and are covered in fuller generality by Neal [2000].

It is worth noting that the sequence of order of appearance labels is not exchangeable; for instance, the first label is always 1. However, the prediction rule for  $Z_N$  given  $(Z_1, \dots, Z_{N-1})$  breaks into two parts: (1) the probability of  $Z_N$  taking a value either in  $\{Z_1, \dots, Z_{N-1}\}$  or a new value and (2) the distribution of  $Z_N$  when it takes a new value. When programming such a sampler, it is often useful to simply encode the sets of unique values, which may be done by retaining any set of labels that induce the correct partition (e.g. integer labels) and separately the set of unique parameter values. Indeed, updating the parameter values and partition block belonging separately can lead to improved mixing of the sampler [MacEachern, 1994].

Similarly, in the feature case, we imagine the following generative model for our data. First, let  $F_N$  be a random feature allocation generated according to the EPPF  $p$ . For the  $k$ th feature block in an order of appearance labeling scheme, assign a random label  $\phi_k \sim H$  to this block for some continuous distribution  $H$ . For each  $n$ , let  $Y_n = \{\phi_k : k \in J_n\}$ , where  $J_n$  is here the set of order of appearance labels of the features to which  $n$  belongs. Finally, as above,

$$X_n \stackrel{\text{indep}}{\sim} \hat{F}(Y_n),$$

where the data likelihood  $\hat{F}$  and parameter distribution  $H$  are again application-specific and where now  $\hat{F}$  depends on the variable-size collection of parameters in  $Y_n$ .

Again, we observe that although the order of appearance label sets are not exchangeable, the sequence  $(Y_n)$  is. This fact allows the formulation of a Gibbs sampler via the observation that the distribution of  $Y_n$  given the remaining elements  $\mathbf{Y}_{-n} := (Y_1, \dots, Y_{n-1}, Y_{n+1}, \dots, Y_N)$  is the same as if we thought of  $Y_n$  as the final,  $N$ th element in a sequence with  $N - 1$  preceding values given by  $\mathbf{Y}_{-n}$ . The full details of such a sampler in the IBP case (Examples 5 and 7) are given by Griffiths and Ghahramani [2006].

As in the partition case, in practice when programming the sampler, it is useful to separate the feature allocation encoding from the feature parameter values. Griffiths and Ghahramani [2006] describe how *left order form* matrices give a convenient representation of the feature allocation in this context.

## 4 Stick lengths

Not every symmetric function defined for an arbitrary number of arguments with values in the unit interval is an EPPF [Pitman, 1995], and not every symmetric function with an additional positive integer argument is an EPPF. For instance, the consistency property in Eq. (1) implies certain additivity requirements for the function  $p$ .

**Example 8** (Not an EPPF). Consider the function  $p$  defined with

$$p(1) = 1, \quad p(1, 1) = 0.1, \quad p(2) = 0.8, \quad \dots$$



From the information above,  $p$  may be further defined so as to be symmetric in its arguments for any number of arguments, but since it does not satisfy  $p(1) = p(1, 1) + p(2)$ , it cannot be an EPPF. ■

**Example 9** (Not an EPPF). Consider the function  $p$  defined with

$$p(N = 1) = 0.9, \quad p(N = 1, 1) = 0.9, \quad p(N = 1, 1, 1) = 0.9, \quad \dots$$

From the information above,  $p$  may be further defined so as to be symmetric in its arguments for any number of arguments after the initial  $N$  argument, but since  $(0!)^{-1}p(N = 1) + (1!)^{-1}p(N = 1, 1) + (2!)^{-1}p(N = 1, 1, 1) > 1$ , it cannot be an EPPF. ■

It therefore requires some care to define a suitable distribution over consistent, exchangeable random feature allocations or partitions using the exchangeable probability function framework.

Since we are working with exchangeable sequences of random variables, it is natural to turn to de Finetti's theorem [De Finetti, 1931] for clues as to how to proceed. De Finetti's theorem tells us that any exchangeable sequence of random variables can be expressed as an independent and identically distributed sequence when conditioned on an underlying random *mixing measure*. While this theorem may seem difficult to apply directly to, e.g., exchangeable partitions, it may be applied more naturally to an exchangeable sequence of numbers derived from a sequence of partitions. The argument below is due to Aldous [1985].

Suppose that  $(\Pi_n)$  is an exchangeable, consistent sequence of random partitions. Consider the  $k$ th partition block to appear according to an order of appearance labeling scheme, and give this block a new *random* label  $\phi_k \sim \text{Unif}([0, 1])$  such that each random label is drawn independently from the rest. This construction is the same as the one used for parameter generation in Section 3.4, and  $(\Pi_n)$  is exchangeable by the same arguments used there.

If we apply de Finetti's theorem to the sequence  $(Z_n)$  and note that  $(Z_n)$  has at most countably many different values, we see that there exists some random sequence  $(\rho_k)$  such that  $\rho_k \in (0, 1]$  for all  $k$  and, conditioned on the frequencies  $(\rho_k)$ ,  $(Z_n)$  has the same distribution as iid draws from  $(\rho_k)$ . In this description, we have brushed over technicalities associated with partition blocks that contain only one index even as  $N \rightarrow \infty$  (which may imply  $\sum_k \rho_k < 1$ ).

But if we assume that every partition block eventually contains at least two indices, we can achieve an exchangeable partition of  $[N]$  as follows. Let  $(\rho_k)$  represent a sequence of values in  $(0, 1]$  such that  $\sum_{k=1}^{\infty} \rho_k \stackrel{a.s.}{=} 1$ . Draw  $Z_n \stackrel{iid}{\sim} \text{Discrete}((\rho_k)_k)$ . Let  $\Pi_N$  be the induced partition given  $(Z_n)_{n=1}^N$ . Exchangeability follows from the iid draws, and consistency follows from the induced partition construction.

When the frequencies  $(\rho_k)$  are thought of as subintervals of the unit interval, i.e. a partition of the unit interval, they are collectively called *Kingman's paint-box* [Kingman, 1978]. As another naming convention, we may think of the unit interval as a *stick* [Ishwaran and James, 2001]. We partition the unit interval

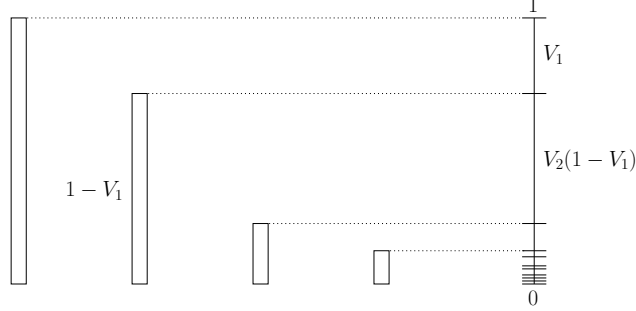


Figure 3: An illustration of how stick-breaking divides the unit interval into a sequence of probabilities. The stick proportions  $(V_1, V_2, \dots)$  determine what fraction of the remaining stick is appended to the probability sequence at each round.

by breaking it into various *stick lengths*, which represent the frequencies of each partition block.

A similar construction can be seen to yield exchangeable, consistent random feature allocations. In this case, let  $(\xi_k)$  represent a sequence of values in  $(0, 1]$  such that  $\sum_{k=1}^{\infty} \xi_k \stackrel{a.s.}{<} \infty$ . We generate feature collections independently for each index as follows. Start with  $Y_n = \emptyset$ . For each feature  $k$ , add  $k$  to the set  $Y_n$ , independently from all other features, with probability  $\xi_k$ . Let  $F_N$  be the induced feature allocation given  $(Y_n)_{n=1}^N$ . Exchangeability of  $F_N$  follows from the iid draws of  $Y_n$ , and consistency follows from the induced feature allocation construction. The finite sum constraint ensures each index belongs to a finite number of features a.s.

It remains to specify a distribution on the partition or feature frequencies. The frequencies cannot be iid due to the finite summation constraint in either case. In the partition case, any infinite set of frequencies cannot even be independent since the summation is fixed to one. One scheme to ensure summation to unity is called *stick-breaking* [McCloskey, 1965, Patil and Taillie, 1977, Sethuraman, 1994, Ishwaran and James, 2001]. In stick-breaking, the stick lengths are obtained by recursively breaking off proportions of the unit interval to return as the atoms  $\rho_1, \rho_2, \dots$  (cf. Figure 3). In particular, we generate the stick-breaking proportions  $V_1, V_2, \dots$  as  $[0, 1]$ -valued random variables. Then  $\rho_1$  is the first proportion  $V_1$  times the initial stick length 1; hence  $\rho_1 = V_1$ . Recursively, after  $k$  breaks, the remaining length of the initial unit interval is  $\prod_{j=1}^k (1 - V_j)$ . And  $\rho_{k+1}$  is the proportion  $V_{k+1}$  of the remaining stick; hence  $\rho_{k+1} = V_{k+1} \prod_{j=1}^k (1 - V_j)$ .

The stick-breaking construction yields  $\rho_1, \rho_2, \dots$  such that  $\rho_k \in [0, 1]$  for each  $k$  and  $\sum_{k=1}^{\infty} \rho_k \leq 1$ . If the  $V_k$  do not decay too rapidly, we will have  $\sum_{k=1}^{\infty} \rho_k \stackrel{a.s.}{=} 1$ . In particular, the partition block proportions  $\rho_k$  sum to unity a.s. iff there is no remaining stick mass:  $\prod_{k=1}^{\infty} (1 - V_k) \stackrel{a.s.}{=} 0$ .

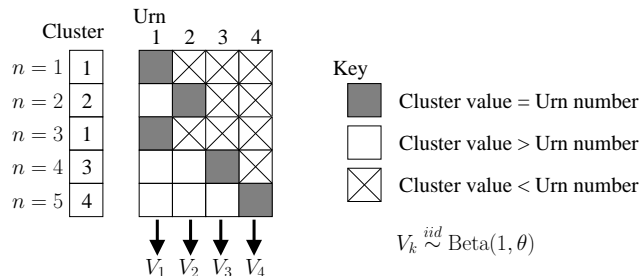


Figure 4: An illustration of the Polya urn proof that Dirichlet process stick-breaking gives the underlying partition block frequencies for a Chinese restaurant process model. The  $k$ th column in the central matrix corresponds to an tallying of when the  $k$ th table is chosen (gray), when a table of index larger than  $k$  is chosen (white), and when an index smaller than  $k$  is chosen ( $\times$ ). If we ignore the  $\times$  tallies, the gray and white tallies can be modeled as balls drawn from a Polya urn. The limiting frequency of gray balls in each column is shown below the matrix.

We often make the additional, convenient assumption that the  $V_k$  are independent. In this case, a necessary and sufficient condition for  $\sum_{k=1}^{\infty} \rho_k \stackrel{a.s.}{=} 1$  is  $\sum_{k=1}^{\infty} \mathbb{E}[\log(1 - V_k)] = -\infty$  [Ishwaran and James, 2001]. When the  $V_k$  are independent and of a canonical distribution, they are easily simulated. Moreover, if we assume that the  $V_k$  are such that the  $\rho_k$  decay sufficiently rapidly in  $k$ , one strategy for simulating a stick-breaking model is to ignore all  $k > K$  for some fixed, finite  $K$ . This approximation is known as truncation [Ishwaran and James, 2001]. It is fortuitously the case that in some models of particular interest, such useful assumptions fall out naturally from the model construction (e.g. Examples 10 and 11).

**Example 10** (Chinese restaurant process). In the original result due to de Finetti [De Finetti, 1931], the exchangeable random variables were zero/one-valued and the mixing measure was a distribution on a single frequency so that the outcomes were conditionally Bernoulli. We will find a similar result in obtaining the stick-breaking proportions for the Chinese restaurant process random partition model here.

We can construct a sequence of binary-valued random variables by dividing the customers in the CRP who are sitting at the first table from the rest; color the former collection of customers gray and the latter collection of customers white. Then, we see that the first customer must be colored gray. And thus we begin with a single gray customer and no white customers. This binary valuation for the first table in the CRP is illustrated by the first column in Figure 4.

At this point, it is useful to recall the Polya urn construction [Pólya, 1930, Freedman, 1965], whereby an urn starts with  $G_0$  gray balls and  $W_0$  white balls. At each round  $N$ , we draw a ball from the urn, replace it, and add  $\kappa$  of the

same color of ball to the urn. At the end of the round, we have  $G_N$  gray balls and  $W_N$  white balls. Despite the urn metaphor, the number of balls need not be an integer at any time. By checking the CRP Eq. (2), we can see that the coloring of the gray/white customer matrix assignments starting with the second customer has the same distributions as a sequence of balls from a Polya urn as a Polya urn with  $G_{1,0} = 1$  initial gray balls,  $W_{1,0} = \theta$  initial white balls, and  $\kappa_1 = 1$  replacement balls. Let  $G_{1,N}$  and  $W_{1,N}$  represent the numbers of gray and white balls, respectively, in the urn after  $N$  rounds. The important fact about the Polya urn we use here is that there exists some  $V \sim \text{Beta}(G_0/\kappa, W_0/\kappa)$  such that  $\kappa^{-1}(G_{N+1} - G_N) \stackrel{iid}{\sim} \text{Bern}(V)$  for all  $N$ . In this particular CRP case, then,  $G_{1,N+1} - G_{1,N}$  is one if a customer sits at the first table, and  $G_{1,N+1} - G_{1,N} \stackrel{iid}{\sim} \text{Bern}(V_1)$  with  $V_1 \sim \text{Beta}(1, \theta)$ .

We now look at the sequence of customers who sit at the second and subsequent tables. That is, we condition on customers not sitting at the first table or equivalently on the sequence with  $G_{1,N+1} - G_{1,N} = 0$ . Again, we have that the first customer sits at the second table, by the CRP construction. Now let customers at the second table be colored gray and customers at the third and later tables be colored white. This valuation is illustrated in the second column in Figure 4; each  $\times$  in the figure denotes a data point where the first partition block is chosen and therefore the current Polya urn is not in play. As before, we begin with one gray customer and no white customers. We can check the CRP Eq. (2) to see that customer coloring once more proceeds according to a Polya urn scheme with  $G_{2,0} = 1$  initial gray balls,  $W_{2,0} = \theta$  initial white balls, and  $\kappa_2 = 1$  replacement balls. Thus, contingent on a customer not sitting at the first table, the  $N$ th customer sits at the second table with iid distribution  $\text{Bern}(V_2)$  with  $V_2 \sim \text{Beta}(1, \theta)$ . And  $V_2$  is independent of  $V_1$ .

The argument just outlined proceeds recursively to show us that the  $N$ th customer, conditional on not sitting at the first  $K - 1$  tables for  $K \geq 1$ , sits at the  $K$ th table with iid distribution  $\text{Bern}(V_K)$  and  $V_K \sim \text{Beta}(1, \theta)$  with  $V_K$  independent of the previous  $(V_1, \dots, V_{K-1})$ .

Combining these results, we see that we have the following construction for the customer seating patterns. The  $V_k$  are distributed independently and identically according to  $\text{Beta}(1, \theta)$ . The probability  $\rho_K$  of sitting at the  $K$ th table is the probability of not sitting at the first  $K - 1$  tables, conditional on not sitting at the previous table, times the conditional probability of sitting at the  $K$ th table:  $\rho_K = \left[ \prod_{k=1}^{K-1} (1 - V_k) \right] \cdot V_K$ . Finally, with the vector of table frequencies  $(\rho_k)$ , each customer sits independently and identically at the corresponding vector of tables according to these frequencies. This process is summarized here:

$$\begin{aligned} V_k &\stackrel{iid}{\sim} \text{Beta}(1, \theta) \\ \rho_K &:= V_K \prod_{k=1}^K (1 - V_k) \\ Z_n &\stackrel{iid}{\sim} \text{Discrete}((\rho_k)_k). \end{aligned} \tag{11}$$

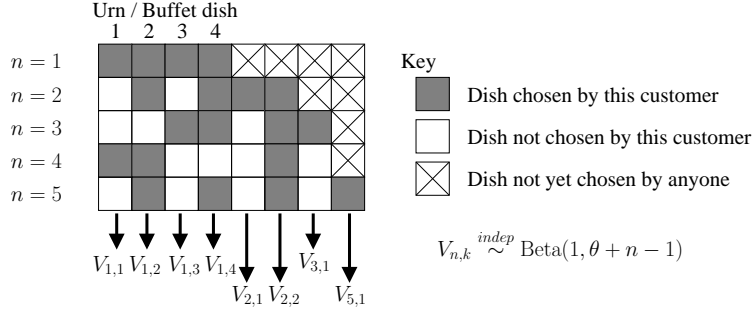


Figure 5: Illustration of the proof that the frequencies of features in the Indian buffet process are given by beta random variables. For each feature, we can construct a sequence of zero/one variables by tallying whether (gray, one) or not (white, zero) that feature is represented by the given data point. Before the first time a feature is chosen, we mark it with an  $\times$ . Each column sequence of gray and white tallies, where we ignore the  $\times$  marks, forms a Polya urn with limiting frequencies shown below the matrix.

■

The feature case is easier. Since it does not require the frequencies to sum to one, the random frequencies can be independent so long as they have an a.s. finite sum.

**Example 11** (Indian buffet process). We use a similar urn approach to the CRP case to recover stick lengths in the Indian buffet process.

Recall that on the first round of the Indian buffet process,  $K_1^+ \sim \text{Pois}(\gamma)$  features are chosen to contain index  $n$ . Consider one of the features, labeled  $k$ . Each future data point  $N$  belongs to this feature with probability  $M_{N,k}/(\theta + N - 1)$ . Thus, we can model the sequence after the first data point as a Polya urn of the sort encountered in Example 10 with initially  $G_{k,0} = 1$  gray balls,  $W_{k,0} = \theta$  white balls, and  $\kappa_k = 1$  replacement balls. As we have seen, there exists a random variable  $V_k \sim \text{Beta}(1, \theta)$  such that representation of this feature by data point  $N$  is chosen, iid across all  $N$ , as  $\text{Bern}(V_k)$ . Since the Bernoulli draws conditional on previous draws are independent across all  $k$ , the  $V_k$  are likewise independent of each other; this fact is also true for  $k$  in future rounds. Draws according to such an urn are illustrated in each of the first four columns of the matrix in Figure 5.

Now consider any round  $n$ . According to the IBP construction,  $K_n^+ \sim \text{Pois}(\gamma\theta/(\theta+n-1))$  new features are chosen to include index  $n$ . Each future data point  $N$  represents feature  $k$  among these features with probability  $M_{N,k}/(\theta + N - n)$ . In this case, we can model the sequence after the  $n$ th data point as a Polya urn with  $G_{k,0} = 1$  initial gray balls,  $W_{k,0} = \theta + n - 1$  initial white balls, and  $\kappa_k = 1$  replacement balls. So there exists a random variable  $V_k \sim \text{Beta}(1, \theta + n - 1)$  such that representation of feature  $k$  by data point  $N$  is chosen,

iid across all  $N$ , as  $\text{Bern}(V_k)$ .

Finally, then, we have the following generative model for the feature allocation [Thibaux and Jordan, 2007].

$$\begin{aligned}
K_n^+ &\overset{\text{indep}}{\sim} \text{Pois}\left(\frac{\gamma\theta}{\theta + n - 1}\right) \\
K_n &= K_{n-1} + K_n^+ \\
V_k &\overset{\text{indep}}{\sim} \text{Beta}(1, \theta + n - 1), \quad k = K_{n-1} + 1, \dots, K_n \\
I_{n,k} &\overset{\text{indep}}{\sim} \text{Bern}(V_k), \quad k = 1, \dots, K_n
\end{aligned} \tag{12}$$

$I_{n,k}$  is an indicator random variable for whether feature  $k$  contains index  $n$ . The collection of features to which index  $n$  belongs,  $Y_n$ , is the collection of features  $k$  with  $I_{n,k} = 1$ . ■

## 4.1 Inference

As we have seen above, the exchangeable probability functions of Section 3 are the marginal distributions of the partitions or feature allocations generated according to stick-length models with the stick lengths integrated out. It has been proposed that including the stick lengths in MCMC samplers of these models will improve mixing [Ishwaran and Zarepour, 2000]. While it is impossible to sample the countably infinite set of partition block or feature frequencies in these models (cf. Examples 10 and 11), a number of ways of getting around this difficulty have been investigated. Ishwaran and Zarepour [2000] examine two separate finite approximations to the full CRP stick length model; one uses a parametric approximation to the full infinite model, and the other creates a truncation by setting the stick break at some fixed size  $K$  to be 1:  $V_K = 1$ . However, retrospective sampling [Papaspiliopoulos and Roberts, 2008] and slice sampling [Walker, 2007] can both be used to avoid any approximations and deal instead directly with the full model.

While our inference discussion thus far has focused on MCMC sampling as a means of approximating the posterior distribution of either the block assignments or both the block assignments and stick lengths, including the stick lengths in a posterior analysis facilitates a different posterior approximation; in particular, *variational methods* can be used to approximate the posterior by minimizing some sense of distance to the posterior over a family of potential approximating distributions [Jordan et al., 1999]. The practicality and, indeed, speed of these methods in the case of stick-breaking for the CRP (Example 10) have been demonstrated by Blei and Jordan [2006].

A number of different models for the stick lengths corresponding to the features of an IBP (Example 11) have been discovered. The distributions described in Example 11 are covered by Thibaux and Jordan [2007], who build on work from Hjort [1990], Kim [1999]. A special case of the IBP is examined by Teh et al. [2007], who detail a slice sampling algorithm for sampling from the posterior of the stick lengths and feature assignments. Yet another stick length

model for the IBP is explored by Paisley et al. [2010], who show how to apply variational methods to approximate the posterior of their model.

Stick length modeling has the further advantage of allowing inference in cases where it is not straightforward to integrate out the underlying stick lengths to obtain a tractable exchangeable probability function.

## 5 Subordinators

An important point to reiterate about the labels  $Z_n$  and label collections  $Y_n$  is that when we use the order of appearance labeling scheme for partition or feature blocks described above, the random sequences  $(Z_n)$  and  $(Y_n)$  are not exchangeable. Often, however, we would like to make use of special properties of exchangeability when dealing with these sequences. For instance, if we use Markov Chain Monte Carlo to sample from the posterior distribution of a partition (cf. Section 3.4), we might want to Gibbs sample  $Y_n$  given  $\{Y_m\} \setminus Y_n$ . This sampling is particularly easy in some cases [Neal, 2000] if we can treat  $Y_n$  as the last random variable in the sequence, but this treatment requires exchangeability.

A way to get around this dilemma was suggested by Aldous [1985] and appeared above in our motivation for using stick lengths. Namely, we assign to the  $k$ th partition block a uniform random label  $\phi_k \sim \text{Unif}([0, 1])$ ; analogously, we assign to the  $k$ th feature a uniform random label  $\phi_k \sim \text{Unif}([0, 1])$ . We can see that in both cases, all of the labels are a.s. distinct. Now, in the partition case, let  $Z_n$  be the uniform random label of the partition block to which  $n$  belongs. And in the feature case, let  $Y_n$  be the (finite) set of uniform random feature labels for the features to which  $n$  belongs. We can recover the partition or feature allocation as the induced partition or feature allocation by grouping indices assigned to the same label. Moreover, as discussed above, we now have that each of  $(Z_n)$  and  $(Y_n)$  is an exchangeable sequence.

If we form partitions or features according to the stick length constructions detailed in Section 4, we know that each unique partition or feature label  $\phi_k$  is associated with a frequency  $\xi_k$ . We can imagine this association in the form of a random measure:

$$\mu = \sum_{k=1}^{\infty} \xi_k \delta_{\phi_k}. \quad (13)$$

In the partition case,  $\sum_k \xi_k = 1$ , so the random measure is a random probability measure, and we may draw  $Z_n \stackrel{iid}{\sim} \mu$ . In the feature case, the weights have a finite sum but do not necessarily sum to one. In the feature case, we draw  $Y_n$  by including each  $\phi_k$  for which  $\text{Bern}(\xi_k)$  yields a draw of 1.

Another way to codify the random measure in Eq. (13) is as a monotone increasing stochastic process on  $[0, 1]$ . Let

$$T_s = \sum_{k=1}^{\infty} \xi_k \mathbb{1}\{\phi_k \leq s\}.$$

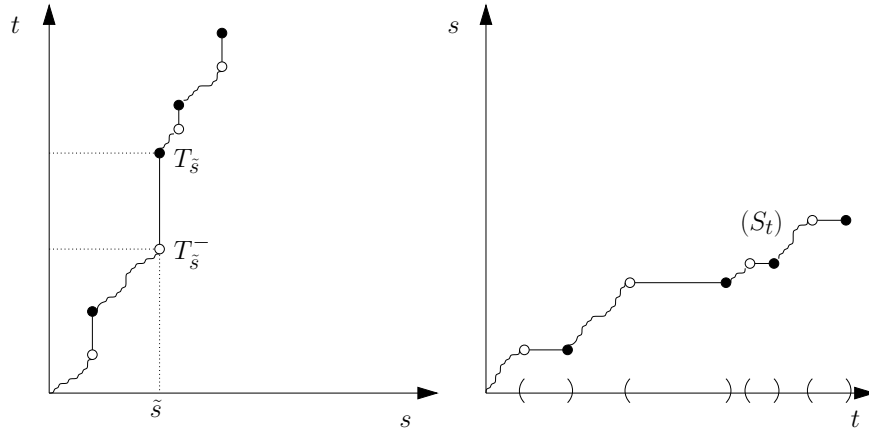


Figure 6: *Left*: The sample path  $(T_s)$  of a subordinator.  $T_{\tilde{s}}^-$  is the left limit of  $(T_s)$  at  $s = \tilde{s}$ . *Right*: The right-continuous inverse  $(S_t)$  of a subordinator. The open intervals along the  $t$  axis correspond to the jumps of the subordinator  $(T_s)$ .

Then the atoms of  $\mu$  are in one-to-one correspondence with the jumps of the process  $T$ .

Taking this increasing random function approach gives us another means of choosing distributions for the weights  $\xi_k$ . We have already seen that these cannot be iid distributed due to the finite summation condition—or even the normalized version of iid random variables in the partition case. However, we will see that if we specify that the *increments* of a monotone, increasing stochastic process are independent and stationary, then we can use the jumps of that function as the atoms in our random measure for partitions or features.

**Definition 12.** A *subordinator* [Bochner, 1955, Bertoin, 1998, 2004] is a stochastic process  $(T_s, s \geq 0)$  that has

- Non-negative, non-decreasing paths (a.s.)
- C  dl  g paths (i.e., paths that are right-continuous with left limits)
- Stationary, independent increments.

For our purposes, wherein the subordinator values will ultimately correspond to (perhaps scaled) probabilities, we will assume the subordinator takes values in  $[0, \infty)$  though alternative ranges with a sense of ordering are possible.

Subordinators are of interest to us because not only do they exhibit the stationary, independent increments property but they can always be decomposed into two components: a deterministic *drift* component and a Poisson point process [Bertoin, 1998]:



**Theorem 13.** *Every subordinator  $(T_s, s \geq 0)$  can be written as*

$$T_s = cs + \sum_{k=1}^{\infty} \xi_k \delta_{\phi_k \leq s}$$

*for some constant  $c \geq 0$  and where  $\{(\xi_k, \phi_k)\}_k$  is the countable set of points of a Poisson process with intensity  $\Lambda(d\xi) d\phi$ , where  $\Lambda$  is a Lévy measure, i.e.*

$$\int_0^{\infty} (1 \wedge \xi) \Lambda(d\xi) < \infty.$$

In particular, then, if a subordinator is finite at time  $t$ , the jumps of the subordinator up to  $t$  may be used as feature block frequencies if they have support in  $[0, 1]$ . Or, in general, the normalized jumps may be used as partition block frequencies. In either case, we have substituted the condition of independent and identical distribution with a more natural continuous-time analogue: independent, stationary intervals.

Just as the Laplace transform of a positive random variable characterizes the distribution of that random variable, so does the Laplace transform of the subordinator—which is a positive random variable at any fixed time point—describe this stochastic process [Bertoin, 1998, 2004].

**Theorem 14** (Lévy-Khinchin formula for subordinators). *If  $(T_s, s \geq 0)$  is a subordinator, then*

$$\mathbb{E}(e^{-\lambda T_s}) = e^{-\Psi(\lambda)s} \quad (14)$$

*with*

$$\Psi(\lambda) = c\lambda + \int_0^{\infty} (1 - e^{-\lambda\xi}) \Lambda(d\xi), \quad (15)$$

*where  $c \geq 0$  is called the drift constant and  $\Lambda$  is a non-negative, Lévy measure on  $(0, \infty)$ .*

The Laplace transform is called the *Laplace exponent* in this context. We note that a subordinator is characterized by its drift constant and Lévy measure.

Using subordinators for feature allocation modeling is particularly easy; since the jumps of the subordinators are formed by a Poisson point process, we can use Poisson process methodology to find the stick lengths and EFPPF that result when we choose each index’s feature belonging according to Bernoulli draws at the jumps of the subordinator.

**Example 15** (Indian buffet process). So far, we have found a collection of stick lengths to represent the featural frequencies for the IBP (Eq. (12) of Example 11 in Section 4). To see the connection to subordinators, we start from the *beta process subordinator* [Kim, 1999] with zero drift ( $c = 0$ ) and Lévy measure

$$\Lambda(d\xi) = \gamma \theta \xi^{-1} (1 - \xi)^{\theta-1} d\xi. \quad (16)$$

We will see that the mass parameter  $\gamma > 0$  and concentration parameter  $\theta > 0$  are the same as those introduced in Example 5 and continued in Example 11.

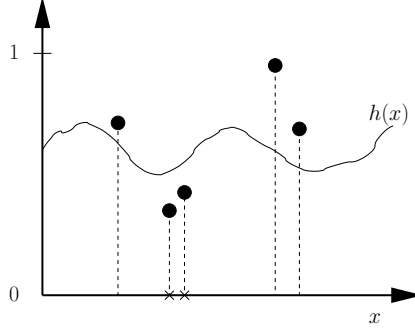


Figure 7: An illustration of Poisson thinning. The  $x$ -axis values of the filled black circles, emphasized by dotted lines, are generated according to a Poisson process. The  $[0, 1]$ -valued function  $h(x)$  is arbitrary. The vertical axis values of the points are uniform draws in  $[0, 1]$ . The “thinned” points are the collection of  $x$ -axis values corresponding to vertical axis values below  $h(x)$  and are denoted with a  $\times$  symbol.

**Theorem 16.** *The size-biased jumps of the beta process subordinator with Lévy density given by Eq. (16) have the same distribution as the IBP stick lengths given by Eq. (12) in Example 5, namely:*

$$\begin{aligned}
K_n^+ &\stackrel{\text{indep}}{\sim} \text{Pois}\left(\frac{\gamma\theta}{\theta + n - 1}\right) \\
K_n &= K_{n-1} + K_n^+ \\
V_k &\stackrel{\text{indep}}{\sim} \text{Beta}(1, \theta + n - 1), \quad k = K_{n-1} + 1, \dots, K_n \\
I_{n,k} &\stackrel{\text{indep}}{\sim} \text{Bern}(V_k), \quad k = 1, \dots, K_n.
\end{aligned}$$

*Proof.* Recall the following fact about Poisson thinning [Kingman, 1993], illustrated in Figure 7. Suppose that a Poisson point process with rate measure  $\lambda$  generates points with values  $x$ . Then suppose that, for each such point  $x$ , we keep it with probability  $h(x) \in [0, 1]$ . The resulting set of points is also a Poisson point process, now with rate measure  $\lambda'(A) = \int_A \lambda(dx)h(x) dx$ .

Consider again the proposal to generate feature membership from a subordinator by taking Bernoulli draws at each of its jumps with success probability equal to the jump size. Since every jump has strictly positive size, the feature associated with each jump will eventually score a Bernoulli success for some index  $n$  with probability one. Therefore, we can enumerate all jumps of the process by first enumerating all features in which index 1 appears, then all features in which index 2 appears but not index 1, and so on; at the  $n$ th iteration, we enumerate all features in which index  $n$  appears but not previous indices.

We prove Theorem 16 recursively. Define the measure

$$\mu_n(d\xi) := \gamma\theta\xi^{-1}(1 - \xi)^{\theta+n-1} d\xi,$$

so that  $\mu_0$  is the beta process Lévy measure  $\Lambda$  in Eq. (16). We make the recursive assumption that  $\mu_n$  is distributed as the beta process measure without atoms corresponding to features chosen on the first  $n$  iterations.

There are two parts to proving Theorem 16. First, we show that, on the  $n$ th iteration, the number of features chosen and the distribution of the corresponding atom weights agree with Eq. (12). Second, we check that the recursion assumption holds.

For the first part, note that on the  $n$ th round we choose features with probability equal to their atom weight. So we form a thinned Poisson process with rate measure  $\xi \cdot \mu_{n-1}(d\xi)$ . This rate measure has total mass

$$\int_{\xi} \xi \cdot \mu_{n-1}(d\xi) = \gamma \frac{\theta}{\theta + n - 1} =: \gamma_{n-1}.$$

So the number of features chosen is Poisson-distributed with mean  $\gamma\theta(\theta + n - 1)^{-1}$ , as desired. And the atom weights have distribution equal to the normalized rate measure

$$\gamma_{n-1}^{-1} \xi \cdot \gamma\theta\xi^{-1}(1 - \xi)^{\theta+n-1} d\xi = \text{Beta}(\xi|1, \theta + n - 1)d\xi,$$

as desired.

Finally, to check the recursion assumption, we note that those sticks that remained were chosen for having Bernoulli failure draws; i.e., they were chosen with probability equal to one minus their atom weight. So the thinned rate measure for the next round is

$$(1 - \xi) \cdot \gamma\theta\xi^{-1}(1 - \xi)^{\theta+(n-1)-1} d\xi,$$

which is just  $\mu_n$ . □

The form of the EPPF of the feature allocation generated from the beta process subordinator follows immediately from the stick length distributions we have just derived by the discussion in Example 11 in Section 4. ■

We see from the previous example that feature allocation stick lengths and EPPFs can be obtained in a straightforward manner using the Poisson process representation of the jumps of the subordinator. Partitions, however, are not as easy to analyze, principally due to the fact that the subordinator jumps must first be normalized to obtain a probability measure on  $[0, 1]$ , rather than just a random measure with finite total mass. We must compute the stick lengths and EPPF using partition block frequencies from these normalized jumps instead of the direct subordinator jumps.

In the EPPF case, we make use of a result that gives us the exchangeable probability function as a function of the Laplace exponent. Though we do not derive this formula here, its derivation can be found in Pitman [2003]; the proof relies on, first, calculating the joint distribution of the subordinator jumps and partition generated from the normalized jumps and, second, integrating out the subordinator jumps to find the partition marginal.

**Theorem 17.** Let  $(\Pi_n)$  be a consistent set of exchangeable partitions. For each exchangeable partition  $\pi_N = \{A_1, \dots, A_K\}$  of  $[N]$  with  $N_k := |A_k|$  for each  $k$ ,

$$\begin{aligned} \mathbb{P}(\Pi_N = \{A_1, \dots, A_K\}) &= p(N_1, \dots, N_K) \\ &= \frac{(-1)^{N-K}}{(N-1)!} \int_0^\infty \lambda^{N-1} e^{-\Psi(\lambda)} \prod_{k=1}^K \Psi^{(N_k)}(\lambda) d\lambda \end{aligned} \quad (17)$$

**Example 18** (Chinese restaurant process). We start by introducing the *gamma process*, a subordinator which we will see below generates the Chinese restaurant process EPPF. The gamma process has Laplace exponent  $\Phi(\lambda)$  (Eq. (14)) characterized by

$$c = 0, \quad \text{and} \quad \Lambda(d\xi) = \theta \xi^{-1} e^{-b\xi} d\xi \quad (18)$$

for  $\theta > 0$  and  $b > 0$  (cf. Eq. (15) in Theorem 14). We will see that  $\theta$  corresponds to the CRP concentration parameter  $\theta$  and that  $b$  is arbitrary and does not affect the partition model.

We calculate the EPPF using Theorem 17.

**Theorem 19.** The EPPF for partition block membership chosen according to the normalized jumps  $\{\rho_k\}$  of the gamma subordinator with parameter  $\theta$  is the CRP EPPF (Eq. (3)).

*Proof.* By Theorem 17, if we can find all order derivatives of  $\Psi$ , we can calculate the EPPF for the partitions generated with frequencies equal to the normalized jumps of this subordinator. The derivatives of  $\Psi$ , which are known to always exist [Bertoin, 2000, Rogers and Williams, 2000], are straightforward to calculate if we begin by noting that, from Eq. (15) in Theorem 14, we have

$$\Psi'(\lambda) = c + \int_0^\infty \xi e^{-\lambda\xi} \Lambda(d\xi).$$

Hence,

$$\Psi'(\lambda) = \int_0^\infty e^{-\lambda\xi} \theta e^{-b\xi} d\xi = \frac{\theta}{\lambda + b}$$

Then simple integration and differentiation yield

$$\begin{aligned} \Psi(\lambda) &= \theta \log(\lambda + b) \\ \Psi^{(n)}(\lambda) &= (-1)^{n-1} \frac{(n-1)! \theta}{(\lambda + b)^n}. \end{aligned}$$

We can substitute these quantities into the general EPPF formula in Eq. (17) of Theorem 17 to obtain

$$\begin{aligned} p(N_1, \dots, N_K) &= \frac{(-1)^{N-K}}{(N-1)!} \int_0^\infty \lambda^{N-1} (\lambda + b)^{-\theta} \prod_{k=1}^K (-1)^{N_k-1} \frac{(N_k-1)! \theta}{(\lambda + b)^{N_k}} d\lambda \end{aligned}$$

$$\begin{aligned}
&= \frac{\theta^K}{(N-1)!} \left[ \prod_{k=1}^K (N_k - 1)! \right] \int_0^\infty \lambda^{N-1} (\lambda + b)^{-N-\theta} d\lambda \\
&= \frac{\theta^K}{(N-1)!} \left[ \prod_{k=1}^K (N_k - 1)! \right] \frac{\Gamma(N)\Gamma(\theta)}{\Gamma(N+\theta)} \\
&= \theta^K \left[ \prod_{k=1}^K (N_k - 1)! \right] \frac{1}{\theta(\theta+1)_{N-1 \uparrow 1}}
\end{aligned}$$

The penultimate line follows from the form of the beta prime distribution, and the last line is the CRP EPPF from Eq. (3), as desired. We note in particular that the parameter  $b$  does not appear in the final EPPF.  $\square$

■

Whenever the Laplace exponent of a subordinator is known, Theorem 17 can similarly be applied to quickly find the EPPF of the partition generated by sampling from the normalized subordinator jumps.

To find the stick lengths, i.e. partition block frequency distributions, from the subordinator representation for a partition, we must find the distributions of the normalized subordinator jumps.

**Example 20** (Chinese restaurant process). We continue with the CRP example.

**Theorem 21.** *The size-biased, normalized jumps  $(\rho_k)$ , i.e. jumps in order of appearance, of the gamma subordinator with concentration parameter  $\theta$  (and arbitrary parameter  $b > 0$ ) have the same distribution as the CRP stick lengths (Eq. (11) of Example 10 in Section 4):*

$$\rho_k = V_k \prod_{j=1}^{k-1} (1 - V_j) \quad \text{for} \quad V_j \stackrel{iid}{\sim} \text{Beta}(1, \theta)$$

*Proof.* First, we introduce some notation. Let  $\tau = \sum_k \xi_k$ , the sum over all of the jumps of the subordinator. Second, let  $\tau_k = \tau - \sum_{j=1}^k \xi_k$ , the total sum minus the first  $k$  elements. Finally, let  $W_k = \tau_k / \tau_{k-1}$  and  $V_k = 1 - W_k$ . Then a simple telescoping of factors shows that  $\rho_k = V_k \prod_{j=1}^{k-1} (1 - V_j)$ :

$$V_k \prod_{j=1}^{k-1} (1 - V_j) = \left(1 - \frac{\tau_k}{\tau_{k-1}}\right) \prod_{j=1}^{k-1} \frac{\tau_j}{\tau_{j-1}} = \frac{\tau_{k-1} - \tau_k}{\tau_0} = \frac{\xi_k}{\tau} = \rho_k$$

It remains to show that the  $V_k$  have the desired distribution. To that end, it is easier to work with the  $W_k$ . We will find the following lemma [Pitman, 2006] useful.

**Lemma 22.** *Let  $\rho$  be the density of  $\Lambda$  with respect to Lebesgue measure. And let  $f$  be the density of the distribution of  $\tau$  with respect to Lebesgue measure. Then*

$$\mathbb{P}(\tau_0 \in dt_0, \dots, \tau_k \in dt_k)$$

$$= f(t_k) dt_k \left( \prod_{j=0}^{k-1} \frac{(t_j - t_{j+1}) \rho(t_j - t_{j+1})}{t_j} dt_j \right)$$

With this lemma in hand, the result follows from a change of variables calculation; we use a bijection between  $\{W_1, \dots, W_k, T\}$  and  $\{\tau_0, \dots, \tau_k\}$  defined by  $\tau_k = \tau \prod_{j=1}^k W_j$ . The determinant of the Jacobian for the transformation to the latter from the former is

$$J = \prod_{j=0}^{k-1} \tau_k,$$

where we note that  $\tau_k$  is a function of  $\tau$  and  $\{W_1, \dots, W_k\}$ . Then

$$\begin{aligned} & \mathbb{P}(W_1 \in dw_1, \dots, W_k \in dw_k, T \in dt_0) \\ &= \mathbb{P}(\tau_0 \in dt_0, \dots, \tau_k \in dt_k) \cdot J \\ &= f(t_k) dt_k \left( \prod_{j=0}^{k-1} (t_j - t_{j+1}) \rho(t_j - t_{j+1}) \right) \end{aligned}$$

In the case of the gamma process, we can read  $\rho(\xi) = \theta \xi^{-1} e^{-b\xi}$  from Eq. (18). The function  $f$  is determined by  $\rho$  and in this case [Pitman, 2006]:

$$f(t) = \text{Ga}(t|\theta, b) = b^{-\theta} \Gamma(\theta)^{-1} t^{\theta-1} e^{-bt}.$$

So

$$\begin{aligned} & \mathbb{P}(W_1 \in dw_1, \dots, W_k \in dw_k, \tau \in dt_0) \\ & \propto t_k^{\theta-1} e^{-bt_0} = t^{\theta-1} e^{-bt} \prod_{j=1}^k w_j^{\theta-1} \end{aligned}$$

Since the distribution factorizes, the  $\{W_k\}$  are independent of each other and of  $\tau$ . Second, we can read off the distributional kernel of each  $W_k$  to establish  $W_k \stackrel{iid}{\sim} \text{Beta}(\theta, 1)$ , from whence it follows that  $V_k \stackrel{iid}{\sim} \text{Beta}(1, \theta)$ .  $\square$

■

## 5.1 Inference

In some sense, we skipped ahead in describing inference in Sections 3.4 and 4.1. There, we made use of the fact that random labels for partitions and features imply exchangeability of the data partition block assignments  $(Z_n)$  and data feature assignments  $(Y_n)$ . In the discussion above, we study the object that associates random uniformly distributed labels with each partition or feature. Assuming the labels come from a uniform distribution rather than a general continuous distribution is a special case of the discussion in Section 3.4, and we defer the general case to the next section (Section 6).

We have seen above that it is particularly straightforward to obtain an EPPF or EFPF formulation, which yields Gibbs sampling steps as described in Section 3.4, when the stick lengths are generated according to a normalized Poisson process in the partition case or a Poisson process in the feature case. Examples 15 and 18 illustrate how to find such exchangeable probability functions. Further, as we have seen the usefulness of the stick representation in inference, Examples 15 and 20 illustrate how stick length distributions may be recovered from the subordinator framework.

## 6 Completely random measures

In the discussion of subordinators above, the jump sizes of the subordinator correspond to the feature frequencies or unnormalized partition frequencies and are the quantities of interest; the locations of the jump sizes are convenient labels that allow the sequence of index assignments  $(Z_1, Z_2, \dots$  in the clustering case or  $Y_1, Y_2, \dots$  in the feature case) to be exchangeable.

However, this labeling retains the same convenient properties as long as the labels are chosen iid from any continuous distribution (not just the uniform distribution), thereby guaranteeing that each partition block or feature has a unique label a.s. Moreover, in typical applications, we wish to associate some parameter with each partition block or feature. In the partition case, we typically model the observed data  $X_n$  indexed by  $n$  as being generated according to some likelihood depending on the parameter corresponding to its partition block assignment. Likewise, in the feature case, we typically model the observed data  $X_n$  indexed by  $n$  as being generated according to some likelihood depending on the collection of parameters corresponding to its collection of feature block assignments (cf. Eq. (10)).

In these cases, it can be useful to suppose that the partition block labels, or feature labels,  $\phi_k$  are not necessarily  $\mathbb{R}_+$ -valued but rather are generated iid according to some continuous distribution  $H$  on a general space  $\Phi$ . Then, whenever  $k$  is the order of appearance partition block label of index  $n$ , we let  $Z_n = \phi_k$ . Similarly, whenever  $k$  is the order of appearance feature label for some feature to which index  $n$  belongs,  $\phi_k \in Y_n$ . Finally, then, we complete the generative model in the partition case by letting  $X_n \stackrel{\text{indep}}{\sim} \hat{F}(Z_n)$  for some distribution function  $\hat{F}$  depending on parameter  $Z_n$ . And in the feature case,  $X_n \stackrel{\text{indep}}{\sim} \hat{F}(Y_n)$ , where now the distribution function  $\hat{F}$  depends on the collection of parameters  $Y_n$ .

When we take the jump sizes  $(\xi_k)$  of a subordinator as the weights of atoms with locations  $(\phi_k)$  drawn iid according to  $H$  as described above, we find ourselves with a *completely random measure*  $\mu$ :

$$\mu = \sum_{k=1}^{\infty} \xi_k \delta_{\phi_k}. \quad (19)$$

A completely random measure is a random measure  $\mu$  such that whenever  $A$

and  $A'$  are disjoint sets, we have that  $\mu(A)$  and  $\mu(A')$  are independent random variables.

To see that changing the atom locations from a subordinator in this way yields a completely random measure, note that Theorem 13 tells us that the subordinator jumps sizes are generated according to a Poisson point process, with some intensity measure  $\nu(d\xi)$ . The Marking Theorem for Poisson processes [Kingman, 1993] in turn yields that the tuples  $\{(\xi_k, \phi_k)\}_k$  are generated according to a Poisson point process with intensity measure  $\nu(d\xi)H(d\phi)$ . By Kingman [1967], whenever the tuples  $\{(\xi_k, \phi_k)\}_k$  are drawn according to a Poisson point process, the measure in Eq. (19) is completely random.

**Example 23** (Dirichlet process). We can form a completely random measure from the gamma process subordinator and a random labeling of the partition blocks. Specifically, suppose that the labels come from a continuous measure  $H$ . Then we generate a completely random measure  $G$  called a *gamma process* [Ferguson, 1973] in the following way:

$$\nu(d\xi \times d\phi) = \theta \xi^{-1} e^{-b\xi} d\xi \cdot H(d\phi) \quad (20)$$

$$\{(\xi_k, \phi_k)\}_k \sim \text{PPP}(\nu) \quad (21)$$

$$G = \sum_{k=1}^{\infty} \xi_k \delta_{\phi_k} \quad (22)$$

Here,  $\text{PPP}(\nu)$  denotes a draw from a Poisson point process with intensity measure  $\nu$ . The parameters  $\theta > 0$  and  $b > 0$  are the same as for the gamma process subordinator. A gamma process draw, along with its generating Poisson point process intensity measure, is illustrated in Figure 8.

The *Dirichlet process* (DP) is the random measure formed by normalizing the gamma process [Ferguson, 1973]. Since the Dirichlet process atom weights sum to one, it cannot be completely random. We can write the Dirichlet process  $D$  generated from the gamma process  $G$  above as:

$$\begin{aligned} \xi_0 &= \sum_{k=1}^{\infty} \xi_k \\ \rho_k &= \xi_k / \xi_0 \\ D &= \sum_{k=1}^{\infty} \rho_k \delta_{\phi_k}. \end{aligned}$$

The random variables  $\rho_k$  have the same distribution as the Dirichlet process sticks (Eq. (11)) or normalized gamma process subordinator jump lengths, as we have seen above (Example 18). ■

Consider sampling points from a Dirichlet process and forming the induced partition of the data indices. Theorem 19 shows us that the distribution of the induced partition is the Chinese restaurant process EPPF.



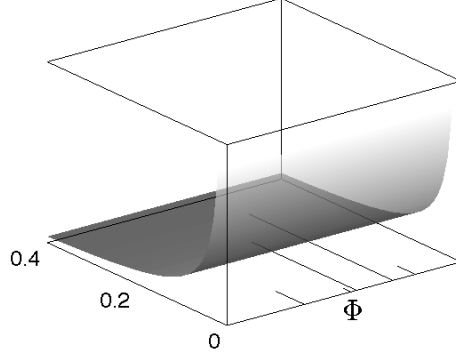


Figure 8: The gray manifold depicts the Poisson point process intensity measure  $\nu$  in Eq. (20) for the choice  $\Phi = [0, 1]$  and  $H$  the uniform distribution on  $[0, 1]$ . The endpoints of the line segments are points drawn from the Poisson point process as in Eq. (21). Taking the first coordinate, which is positive real-valued, as the atom weights, we find the measure  $G$  on  $\Phi$  from Eq. (22) in the bottom plane.

**Example 24** (Beta process). We can form a completely random measure from the beta process subordinator and a random labeling of the feature blocks. If the labels are generated iid from a continuous measure  $H$ , then we say the completely random measure  $B$ , generated as follows, is called a *beta process*.

$$\nu(d\xi \times d\phi) = \gamma \theta \xi^{-1} (1 - \xi)^{\theta-1} d\xi \cdot H(d\phi) \quad (23)$$

$$\{(\xi_k, \phi_k)\}_k \sim \text{PPP}(\nu) \quad (24)$$

$$B = \sum_{k=1}^{\infty} \xi_k \delta_{\phi_k} \quad (25)$$

The beta process, along with its generating intensity measure, is depicted in Figure 9. Then the  $(\xi_k)$  have the same distribution as the beta process sticks (Eq. (12)) or the beta process subordinator jump lengths (Example 15). ■

Now consider sampling a collection of atom locations according to Bernoulli draws from the atom weights of a beta process and forming the induced feature allocation of the data indices. Theorem 16 shows us that the distribution of the induced feature allocation is given by the Indian buffet process EFPF.

## 6.1 Inference

In this section, we finally study the full model first outlined in the context of inference of partition and feature structures in Section 3.4. The partition or feature labels described in this section are the same as the block-specific parameters

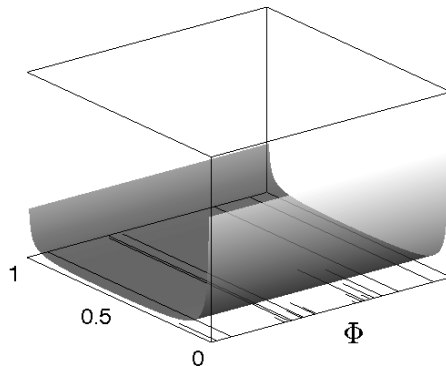


Figure 9: The gray manifold depicts the Poisson point process intensity measure  $\nu$  in Eq. (23) for the choice  $\Phi = [0, 1]$  and  $H$  the uniform distribution on  $[0, 1]$ . The endpoints of the line segments are points drawn from the Poisson point process as in Eq. (24). Taking the first coordinate, which is  $[0, 1]$ -valued, as the atom weights, we find the measure  $B$  on  $\Phi$  from Eq. (25) in the bottom plane.

first described in Section 3.4. Since this section focuses on a generalization of the partition or feature labeling scheme beyond the uniform distribution option encoded in subordinators, inference for the atom weights remains unchanged from Sections 3.4, 4.1, and 5.1.

However, we note that, in the course of inferring underlying partition or feature structures, we are often also interested in inferring the data likelihood parameters that form the partition or feature labels and govern the data distribution within each block. Conditional on the partition or feature structure, such inference is handled as in a normal hierarchical model with fixed dependencies. Namely, the parameter within a particular block may be inferred from the data points that depend on this block as well as the prior distribution for the parameters. Details for the Dirichlet process example inferred via MCMC sampling are provided by MacEachern [1994], Escobar and West [1995], Neal [2000]; Blei and Jordan [2006] work out details for the Dirichlet process using variational methods. In the beta process case, Griffiths and Ghahramani [2006], Teh et al. [2007], Thibaux and Jordan [2007] describe MCMC sampling, and Paisley et al. [2010] describe a variational methods approach.

## 7 Conclusion

In the discussion above, we have pursued a sequential augmentation from (1) simple distributions over partitions and feature allocations in the form of exchangeable probability functions to (2) the representation of stick lengths encoding frequencies of the partition block and feature occurrences to (3) subor-

dinators, which associate random  $\mathbb{R}_+$ -valued labels with each partition block or feature, and finally to (4) completely random measures, which associate a general class of labels with the stick lengths and whose labels we generally use as parameters in likelihood models built from the partition or feature allocation representation.

Along the way, we have focused primarily on two vignettes. We have shown, via these successive augmentations, that the Chinese restaurant process specifies the marginal distribution of the induced partition formed from iid draws from a Dirichlet process, which is in turn a normalized completely random measure. And we have shown that the Indian buffet process specifies the marginal distribution of the induced feature allocation formed by iid Bernoulli draws across the weights of a beta process.

There are many extensions of these ideas that lie beyond the scope of this paper. A number of extensions of the CRP and Dirichlet process exist—in either the EPPF form [Pitman, 1996, Blei and Frazier, 2010], the stick length form [Dunson and Park, 2008], or the random measure form [Pitman and Yor, 1997]. Likewise, extensions of the IBP and beta process have been explored [Teh et al., 2007, Paisley et al., 2010, Broderick et al., 2012].

More generally, the framework above demonstrates how alternative partition and feature allocation models may be constructed—either by introducing different EPPFs [Pitman, 1996, Gnedin and Pitman, 2006] or EFPFs, different stick length distributions [Ishwaran and James, 2001], or different random measures [Wolpert and Ickstadt, 2004].

Finally, we note that expanding the set of combinatorial structures with useful Bayesian priors from partitions to the superset of feature allocations suggests that further such structures might be usefully examined. For instance, the *beta negative binomial process* [Broderick et al., 2011] provides a prior on a generalization of a feature allocation where we allow the features themselves to be multisets; i.e., each index may have non-negative integer multiplicities of features. Models on trees [Adams et al., 2010, McCullagh et al., 2008, Blei et al., 2010], graphs [Li and McCallum, 2006], and permutations [Pitman, 1996] provide avenues for future exploration. And there likely remain further structures to be fitted out with useful Bayesian priors.

## References

- R.P. Adams, Z. Ghahramani, and M.I. Jordan. Tree-structured stick breaking for hierarchical data. *Advances in Neural Information Processing Systems*, 23:19–27, 2010.
- D. Aldous. Exchangeability and related topics. *École d’Été de Probabilités de Saint-Flour XIII1983*, pages 1–198, 1985.
- J. Bertoin. *Lévy Processes*, volume 121. Cambridge University Press, 1998.

- J. Bertoin. Subordinators, Lévy processes with no negative jumps, and branching processes. 2000.
- J. Bertoin. Subordinators: examples and applications. *Lectures on Probability Theory and Statistics*, pages 1–91, 2004.
- D. Blackwell and J. B. MacQueen. Ferguson distributions via Pólya urn schemes. *The Annals of Statistics*, 1(2):353–355, 1973. ISSN 0090-5364.
- D. Blei and P. Frazier. Distance dependent chinese restaurant processes. In *International Conference on Machine Learning*, 2010.
- D.M. Blei and M.I. Jordan. Variational inference for dirichlet process mixtures. *Bayesian Analysis*, 1(1):121–144, 2006.
- D.M. Blei, T.L. Griffiths, and M.I. Jordan. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM (JACM)*, 57(2):7, 2010.
- S. Bochner. *Harmonic analysis and the theory of probability*. University of California press, 1955.
- T. Broderick, L. Mackey, J. Paisley, and M.I. Jordan. Combinatorial clustering and the beta negative binomial process. *Arxiv preprint arXiv:1111.1802*, 2011.
- T. Broderick, M. I. Jordan, and J. Pitman. Beta processes, stick-breaking, and power laws. *Bayesian Analysis*, 7, 2012.
- B. De Finetti. Funzione caratteristica di un fenomeno aleatorio. *Atti della R. Accademia Nazionale dei Lincei, Serie 6.*, 4:251–299, 1931. in Italian.
- D.B. Dunson and J.H. Park. Kernel stick-breaking processes. *Biometrika*, 95(2):307–323, 2008.
- M.D. Escobar. Estimating normal means with a dirichlet process prior. *Journal of the American Statistical Association*, pages 268–277, 1994.
- M.D. Escobar and M. West. Bayesian density estimation and inference using mixtures. *Journal of the american statistical association*, pages 577–588, 1995.
- T. S. Ferguson. A Bayesian analysis of some nonparametric problems. *Annals of Statistics*, 1(2):209–230, 1973.
- D. A. Freedman. Bernard Friedman’s urn. *The Annals of Mathematical Statistics*, 36(3):956–970, 1965.
- S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):721–741, 1984.
- A. Gnedin and J. Pitman. Exchangeable Gibbs partitions and Stirling triangles. *Journal of Mathematical Sciences*, 138(3):5674–5685, 2006.

- T. Griffiths and Z. Ghahramani. Infinite latent feature models and the Indian buffet process. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems 18*, pages 475–482. MIT Press, Cambridge, MA, 2006.
- B. Hansen and J. Pitman. Prediction rules for exchangeable sequences related to species sampling. Technical Report 520, University of California, Berkeley, May 1998.
- N. L. Hjort. Nonparametric bayes estimators based on beta processes in models for life history data. *Annals of Statistics*, 18(3):1259–1294, 1990.
- F. M. Hoppe. Pölya-like urns and the Ewens’ sampling formula. *Journal of Mathematical Biology*, 20(1):91–94, 1984.
- H. Ishwaran and L. F. James. Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96(453):161–173, 2001. ISSN 0162-1459.
- H. Ishwaran and M. Zarepour. Markov chain Monte Carlo in approximate Dirichlet and beta two-parameter process hierarchical models. *Biometrika*, 87(2):371–390, 2000.
- M.I. Jordan, Z. Ghahramani, T.S. Jaakkola, and L.K. Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.
- Y. Kim. Nonparametric Bayesian estimators for counting processes. *Annals of Statistics*, 27(2):562–588, 1999.
- J. F. C. Kingman. Completely random measures. *Pacific Journal of Mathematics*, 21(1):59–78, 1967.
- J. F. C. Kingman. The representation of partition structures. *Journal of the London Mathematical Society*, 2(2):374, 1978.
- J. F. C. Kingman. *Poisson Processes*. Oxford University Press, 1993. ISBN 0198536933.
- W. Li and A. McCallum. Pachinko allocation: Dag-structured mixture models of topic correlations. In *Proceedings of the 23rd international conference on Machine learning*, pages 577–584. ACM, 2006.
- S.N. MacEachern. Estimating normal means with a conjugate style dirichlet process prior. *Communications in Statistics-Simulation and Computation*, 23(3):727–741, 1994.
- J. W. McCloskey. *A model for the distribution of individuals by species in an environment*. PhD thesis, Michigan State University, 1965.

- P. McCullagh, J. Pitman, and M. Winkel. Gibbs fragmentation trees. *Bernoulli*, 14(4):988–1002, 2008.
- R. M. Neal. Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9(2):249–265, 2000.
- J. Paisley, A. Zaas, C. W. Woods, G. S. Ginsburg, and L. Carin. A stick-breaking construction of the beta process. In *International Conference on Machine Learning*, Haifa, Israel, 2010.
- O. Papaspiliopoulos and G.O. Roberts. Retrospective Markov chain Monte Carlo methods for Dirichlet process hierarchical models. *Biometrika*, 95(1):169–186, 2008.
- G. P. Patil and C. Taillie. Diversity as a concept and its implications for random communities. In *Proceedings of the 41st Session of the International Statistical Institute*, pages 497–515, New Delhi, 1977.
- J. Pitman. Exchangeable and partially exchangeable random partitions. *Probability Theory and Related Fields*, 102(2):145–158, 1995.
- J. Pitman. Some developments of the Blackwell-MacQueen urn scheme. *Lecture Notes-Monograph Series*, pages 245–267, 1996.
- J. Pitman. Poisson-Kingman partitions. *Lecture Notes-Monograph Series*, 40:1–34, 2003. ISSN 0749-2170.
- J. Pitman. *Combinatorial stochastic processes*, volume 1875 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 2006. ISBN 978-3-540-30990-1; 3-540-30990-X. doi: 10.1007/b11601500. URL <http://bibserver.berkeley.edu/csp/april05/bookcsp.pdf>.
- J. Pitman and M. Yor. The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *Annals of Probability*, 25:855–900, 1997.
- G. Pólya. Sur quelques points de la théorie des probabilités. *Annales de l’I.H.P.*, 1(2):117–161, 1930.
- L. C. G. Rogers and David Williams. *Diffusions, Markov processes, and martingales. Vol. 1*. Cambridge Mathematical Library. Cambridge University Press, Cambridge, 2000. ISBN 0-521-77594-9. Foundations, Reprint of the second (1994) edition.
- J. Sethuraman. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4(2):639–650, 1994.
- Y.W. Teh, D. Görür, and Z. Ghahramani. Stick-breaking construction for the indian buffet process. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, volume 11, 2007.

- R. Thibaux and M. Jordan. Hierarchical beta processes and the Indian buffet process. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, volume 11, 2007.
- S.G. Walker. Sampling the dirichlet mixture model with slices. *Communications in Statistics—Simulation and Computation*®, 36(1):45–54, 2007.
- R.L. Wolpert and K. Ickstadt. Reflecting uncertainty in inverse problems: A bayesian solution using lévy processes. *Inverse Problems*, 20:1759, 2004.